Theoretical Computer Science Department
Faculty of Mathematics and Computer Science
Jagiellonian University

Various Aspects of Nonrepetitive Sequences

Różnorodne oblicza ciągów bez powtórzeń

Karol Kosiński

PhD Thesis
Supervisor: Jarosław Grytczuk

Kraków 2018

Abstract

The famous Thue theorems assert that there exist arbitrarily long words without squares over a 3-letter alphabet and arbitrarily long words without overlaps over a 2-letter alphabet. We consider two consequences of his researches: nonrepetitive games and squarefree colourings of line arrangements.

The bigger part of the thesis is devoted to a game-theoretic variant of Thue result, where a word is constructed jointly by two players who alternately append letters to the end of an existing word. One of the players (Ann) takes care of avoiding predefined repetitions, while the other one (Ben) tries to force them. Our aim is to characterize the winning strategies for Ann dependent on the size of the alphabet and the kind of the repetitions. In particular, we provide explicit algorithms for avoiding: non-trivial squares over an 8-letter alphabet, overlaps over a 4-letter alphabet, and 5th powers over a binary alphabet.

In the remaining part of the thesis we study the following geometric aspects of Thue problem. Given a set L of lines in the plane and a set P of all intersection points of lines in L, what is the least number of colours needed to colour P so that every line in L is squarefree? What is the least number of colours needed to colour the plane so that every path of the unit distance graph whose vertices are colinear is squarefree? We prove that upper bounds for these numbers are respectively: 405 and 36.

Streszczenie

Słynne twierdzenia Thuego zapewniają o istnieniu dowolnie długich słów bez kwadratów nad alfabetem trzyliterowym oraz dowolnie długich slów bez nakładek nad alfabetem dwuliterowym. Rozważamy dwie konsekwencje jego badań: gry bez powtórzeń oraz bezkwadratowe kolorowania ukladów linii prostych.

Większa część pracy jest poświęcona wariantowi rezultatu Thuego wywodzącemu się z teorii gier, gdzie słowo wynikowe jest konstruowane wspólnie przez dwoje graczy, którzy naprzemiennie dołączają litery na koniec istniejącego słowa. Jeden z graczy (Ania) dba o unikanie predefiniowanych powtórzeń, kiedy drugi z nich (Benek) próbuje je wymusić. Naszym celem jest charakteryzacja strategii wygrywających dla Ani zależnie od wielkości używanego alfabetu i rodzaju powtórzeń. W szczególności dostarczamy formalne algorytmy do unikania: nietrywialnych kwadratów nad alfabetem ośmioliterowym, nakladek nad alfabetem czteroliterowym oraz piątych potęg nad alfabetem binarnym.

W pozostalej części pracy studiujemy następujące geometryczne aspekty zagadnienia Thuego. Mając dany zbiór L prostych na płaszczyźnie i zbiór P wszystkich punktów przecięcia prostych z L, jaka jest najmniejsza liczba kolorów potrzebna do pokolorowania P tak, że każda prosta w L jest bezkwadratowa? Jaka jest najmniejsza liczba kolorów potrzebna do pokolorowania plaszczyzny tak, że każda ścieżka grafu o krawędziach dlugości jednostkowej skladającego się ze współliniowych punktów jest bezkwadratowa? Dowodzimy, że ograniczenia górne na te liczby wynoszą odpowiednio: 405 oraz 36.

Acknowledgements

At first, I would like to thank my advisor, Professor Jarosław Grytczuk, for many combinatorial inspirations and invaluable scientific discussions. The second person worth my recognition is Michał Zmarz, who noticed the potential in algorithmic approach to nonrepetitive games. Without the cooperation with them this thesis would not be possible.

I am also grateful to Robert Mercaş and Dirk Nowotka for developing the subject of the squarefree games that allowed me to find some further interesting results in this area.

Finally, I would like to express my special thanks to students and colleagues from Theoretical Computer Science Department for creating a scientific atmosphere and for the time I spent together with them.

Contents

Introduction

1.1 Outline

A finite word is called a *square* (respectively: *cube*, *mth power*) if it is built of two (three, m) identical consecutive subwords. An *overlap* is a finite word consisting of two identical overlapping subwords.

In 1906 Thue proved that there exist arbitrarily long squarefree words over a 3-letter alphabet, and he continued his research which resulted in finding arbitrarily long overlap-free words over a 2-letter alphabet [6, 33, 34]. These results inspired considerable research leading to emergence of a new branch, Combinatorics on Words, with lots of deep results, intriguing connections and important applications in diverse areas of Mathematics and Computer Science (see [1, 7, 13, 19, 26, 29]).

Combinatorics on Words has also a bond with a molecular biology. A genome of living organisms consists of DNA (deoxyribonucleic acid), a structure built of long chains of four nucleobases: adenine ($\mathtt{A}$), cytosine ($\mathtt{C}$), guanine ($\mathtt{G}$) and thymine ($\mathtt{T}$). From an abstract point of view we consider words of length at least tens of millions over a 4-letter alphabet $\{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}$ and check their properties in order to find subwords like squares, cubes and bigger powers (called *tandem repeats*), related to genome mutations and individual's inherited traits.

In this paper we study many game-theoretic variants of Thue's results – let's call them *nonrepetitive games* and present their general idea. More detailed descriptions of particular games are provided later in the thesis.

Definition 1.1 (General nonrepetitive game)**.** *Two players, Ann and Ben, are collectively building a word by alternately appending letters from a fixed alphabet A at the right end of a previously constructed word. Ben's goal is to create a repetition, while Ann tries to prevent it from happening.*

First of them is a *squarefree game* – the game in which Ben wants to construct a square, but Ann attempts to stop him – introduced by Pegden in [30]. Of course, she cannot prevent *trivial* squares of length 1, since Ben may repeat a letter just picked by Ann in her last move. Sometimes in the thesis we refer to the squarefree game as to a *non-trivial-squarefree game.* Can Ann prevent non-trivial squares while playing arbitrarily long with Ben over some finite alphabet? The answer is positive, as proved by Pegden [30]:

Theorem 1.2 (Pegden 2011). *Ann has a strategy to create arbitrarily long words without trivial squares while playing the squarefree game over a* 37*-letter alphabet.*

The proof is non-constructive as it uses a probabilistic argument based on a special version of the Lovász Local Lemma (see [4]). The bound of 37 was next lowered in [22] by using an *entropy compression* argument – a novel technique inspired by an algorithmic version of the Lovász Local Lemma due to Moser and Tardos [28]:

Theorem 1.3 (Grytczuk, Kozik, Micek 2013). *Ann has a strategy to create arbitrarily long words without trivial squares while playing the squarefree game over a* 6*-letter alphabet.*

However, this method is also non-constructive. In particular, it does not guarantee that there is a strategy with *finite description* for Ann over some finite alphabet that works for any length of the play. More formally:

Problem 1.4. *Does a Turing machine winning the squarefree game against Ben if the length of the play is a part of the input exist?*

This question was actually posed by Pegden in [30], and partially answered there by giving such a strategy over 16 letters assuming Ann knows Ben's next move in advance. Pegden also showed (constructively) the lower bound for the size of an alphabet in the squarefree game:

Theorem 1.5 (Pegden 2011). *Ann does not have a strategy to create arbitrarily long words without trivial squares while playing the squarefree game over a* 3*-letter alphabet.*

The structure of the thesis is as follows: in the latter part od the first chapter we present three title infinite words and their selected properties essential to results in succeeding chapters. Continuing the prelimary material, in the second chapter we introduce a simplified variant of nonrepetitive games called *blind games*, in which Ann cannot see Ben's moves. We investigate which types of repetitions she can avoid while appending consecutive letters from nonrepetitive sequences.

In the third chapter we answer Pegden's question from Problem 1.4 in the affirmative unconditionally, presenting the solution using 8 letters. Moreover, we cosider a variant over a 7-letter alphabet in which squares of length 2 are also permitted.

In more complicated variants of nonrepetitive games, which we call *sparse games*, we allow players to place letters in arbitrary positions of a constructed word. It is easy to see that in the sparse squarefree game Ann has no chance for winning over any alphabet (there is a simple "copy-cat" strategy for Ben to create squares of any length). However, the things look quite different in case of overlaps, so in the fourth chapter we prove that Ann is able to win a sparse overlap-free game (with additional initial assumptions) over a 4-letter alphabet and the bound of 4 in this result is optimal.

The fifth chapter is devoted to nonrepetitive games over a 2-letter alphabet. We show there that Ann has a winning strategy for a 5th-power-free game by considering many cases related to the particular lengths of the powers. Furthermore, we note Ann cannot avoid 4th powers for arbitrarily long words.

After the exhaustive proof in the previous part of the thesis, in the sixth chapter we may rest from the games and study the possibilities of squarefree colourings of points on lines in $\mathbb{R}^2$. We obtain at most 405 colours for intersections of the lines from any finite set. Additionally, our another result is that 36 colours is enough to paint squarefreely the points distant from each other by a unit on an arbitrary line.

Finally, we sum up our results in the seventh chapter and present some questions associated to the nonrepetitive games for which we haven't found the answers yet.

1.2 Nonrepetitive sequences

Let's denote an *empty word* as ε. A set of finite nonempty words over an alphabet A denote as A^+, a set of finite words as $A^* = A^+ \cup \{\varepsilon\}$, a set of infinite words as A^ω, and a set of all words as $A^\infty = A^* \cup A^\omega$. We will often refer to a word w of length n as $w_0 w_1 \dots w_{n-1}, w_i \in A$. If $n = 0$, then $w = \varepsilon$. For a word w and a letter $\alpha \in A$, we denote by

$$|w|_\alpha = |\{0 \le i < n : w_i = \alpha\}|$$

the number of times the letter α appears in w.

In the thesis we use terms *word* and *sequence* interchangeably, despite different meaning of notions subword and subsequence. A *subword* w of $u \in A^\infty$ is a finite word satisfying $u = ywz$, where $y \in A^*, z \in A^\infty$. If $y = \varepsilon$, then w is a *prefix* of u. A *subsequence* w of $u \in A^\infty$ is a sequence (maybe infinite) in which $w_i = u_{k_i}$ where $k_0 < k_1 < \dots$ (an *increasing sequence* of indices). Every subword is a subseqence, but not vice versa. For example, `cat` ($k = \{0, 2, 3\}$) and `rat` ($k = \{1, 2, 3\}$) are subsequences of `crate`, but only `rat` is also a subword ($y = \mathtt{c}, z = \mathtt{e}$).

If $\alpha = u_i \in A$ for $u \in A^\infty$, we say that α is *located at position* i in u. If $w \in A^+$ is a subword of u, then w is located at position i when $w_0 = u_i$. A *distance* between subwords $w, v \in A^+$ of u we define as the difference in positions of w and v in u.

A word $w \in A^+$ is called a *square* if it can be written as $w = xx$, where $x \in A^+$. In the similar way we define a *cube* as a word $w \in A^+$ that can we written as $w = xxx$ and an *mth power* as $w = x^m$ (where $x^0 = \varepsilon$ and $x^{k+1} = x^k x$). Note that a square is the 2nd power and a cube is the 3rd power. A *length of square/cube/mth power* is the length of x. By a *parity of square/cube/mth power* we mean the parity of the length of x. An *overlap* is a word w that can be written as $w = axaxa$, where $a \in A$ is a single letter and $x \in A^*$. In other words: two subwords axa overlaps at the letter a. We use the term *repetition* to describe any n-th power or overlap.

For instance, abacabac is a square of length 4, with $x =$ abac, while abacabaca is an overlap, with $x =$ bac. abacabac is an even square, while abcabc is odd. A word ccc is both an overlap ($x = \varepsilon$) and a cube ($x =$ c) of length 1. babababababa is a repetition in three senses: a square of length 6, a cube of length 4, and a 6th power of length 2.

A word $u \in A^\infty$ is *squarefree* if no subword of u is a square. Definitions of *cubefree*, *overlap-free* and *mth-power-free* words are analogous. Any mth-power-free or overlap-free word is *nonrepetitive*.

A *substitution* $f : A \to B^*$ transforms every letter from A to a word from B^*. Its domain can be easily extended from A to A^* by applying f to each letter of $w \in A^*$ and catenating the resultant words.

Let's consider $f : A \to A^*$. f^0 we regard as an *identity function*. If for each $a \in A : f(a) \neq \varepsilon$ and there is $\alpha \in A : f(\alpha) = \alpha u$ for some $u \in A^+$, then

$$f^{n+1}(\alpha) = f^n(\alpha u) = f^n(\alpha) f^n(u)$$

It means $f^n(\alpha)$ is a prefix of $f^{n+1}(\alpha)$, so $\lim_{i\to\infty} f^i(\alpha)$ exists and we are able to extend the domain of f to A^∞. Additionally:

$$x = \lim_{i\to\infty} f^i(\alpha) \quad \Rightarrow \quad f(x) = x$$

The following theorem was proved by Thue [34].

Theorem 1.6. *Let's define a substitution $\psi : \{0, 1\} \to \{0, 1\}^*$:*

$$0 \to 01$$
$$1 \to 10$$

The sequences $\tau = \lim_{i\to\infty} \psi^i(0)$ and $\hat{\tau} = \lim_{i\to\infty} \psi^i(1)$ are overlap-free.

Notice that τ and $\hat{\tau}$ are built on 2-letter *segments*: 01 and 10. The first symbol of each segment is located at even position. Moreover, if we know one letter from any segment, we can deduce the latter symbol. Here we present a couple of simple properties of τ that we will need in the succeeding chapters.

Proposition 1.7. *$\tau_k = \tau_{2k}$ for every $k \in \mathbb{N}$.*

Proof. Let's define a word $w = \tau_0\tau_1 \cdots \tau_{k-1}$. From $\psi(\tau) = \tau$ and the definition of ψ:

$$\psi(w\tau_k) = \psi(w)\psi(\tau_k) = \psi(w)\tau_k\alpha$$

(where $\tau_k \neq \alpha$) is a prefix of τ containing $2k + 2$ letters (ψ doubles the length). The letter at position $2k$ is τ_k. ∎

Proposition 1.8. *For every even square w which is a subword of τ there is an exactly one shorter square u, also a subword of τ, that $w = \psi(u)$.*

Proof. Let a word w be a square for a certain index i in τ and $k = 2j$ $(j > 0)$:

$$w = xx = \tau_i\tau_{i+1}\cdots\tau_{i+k-1}\tau_{i+k}\tau_{i+k+1}\cdots\tau_{i+2k-1}$$

We consider two cases based on the parity of i. If i is even, then x consists of full 2-letter segments `01` and `10`. From Proposition 1.7 we know that the first letters of the segments:

$$\tau_i\tau_{i+2}\cdots\tau_{i+k-2}\tau_{i+k}\tau_{i+k+2}\cdots\tau_{i+2k-2} = (x_0x_2\cdots x_{k-2})^2$$

build a subword of τ, which is the wanted square u.

Assume that i is odd. Obviously, $\tau_i = x_0 = \tau_{i+k}$. From the fact that $\tau_{i-1}\tau_i$ and $\tau_{i+k-1}\tau_{i+k}$ are segments, we observe $\tau_{i-1} = \tau_{i+k-1} = x_{k-1} = \tau_{i+2k-1}$ and, consequently:

$$\tau_{i-1}\tau_i\tau_{i+1}\cdots\tau_{i+k-1}\tau_{i+k}\tau_{i+k+1}\cdots\tau_{i+2k-1} = (x_{k-1}x_0x_1\cdots x_{k-2})^2x_{k-1}$$

Since $k \geq 2$, we get an overlap in τ – a contradiction. ∎

Proposition 1.9. *τ has squares only of length 2^k and $3 \cdot 2^k$ for $k \in \mathbb{N}$.*

Proof. Let's denote a sentence "τ has squares of length 2^k and $3 \cdot 2^k$" by $T(k)$. We use the mathematical induction on k.

$T(k = 0)$**:** Let's look at the initial symbols of τ to find squares of length 1 and 3:

$$\tau = \mathtt{0\underline{11}01\overline{00}1100\underline{1011}\overline{\underline{01}0010}11001101001}\cdots$$

$T(k) \Rightarrow T(k+1)$**:** $\psi(\tau) = \tau$, so if $\psi^k(uu) = \psi^k(u)\psi^k(u)$ is a square of length 2^k or $3 \cdot 2^k$ in τ, where $u \in \{\mathtt{0}, \mathtt{1}, \mathtt{010}, \mathtt{101}\}$, then

$$\psi(\psi^k(uu)) = \psi^{k+1}(uu) = \psi^{k+1}(u)\psi^{k+1}(u)$$

is a square of length $2 \cdot 2^k = 2^{k+1}$ or $2 \cdot 3 \cdot 2^k = 3 \cdot 2^{k+1}$, and from the definition of ψ every its application doubles the length of words.

Therefore $T(k)$ is true for every $k \in \mathbb{N}$. Proposition 1.8 implies that every even square is a result of a single or a multiple application of ψ on odd squares. To complete the proof we also have to show the absence of odd squares of length at least 5 in τ.

Assume the existence of index i that $xx = \tau_i\tau_{i+1}\cdots\tau_{i+2p-1}$ is a square in τ of length $p = 2j + 1$ $(j \geq 2)$. We consider two cases based on the parity of i.

If i is even, xx, as a subword of τ, consists only of full 2-letter segments `01` and `10`. It means that $|xx|_{\mathtt{0}} = |xx|_{\mathtt{1}}$, while we have $|x|_{\mathtt{0}} \neq |x|_{\mathtt{1}}$, because p is odd. Hence, a contradiction.

Thus i is odd. Without loss of generality, let $\tau_i = x_0 = \tau_{i+p} = \mathtt{0}$. Because p is also odd, $\tau_{i+p}\tau_{i+p+1}$ is a 2-letter segment, which implies $\tau_{i+1} = x_1 = \tau_{i+p+1} = \mathtt{1}$. $\tau_{i+1}\tau_{i+2}$ is also a 2-letter segment, so $\tau_{i+p+2} = x_2 = \tau_{i+2} = \mathtt{0}$. After two more iterations we obtain the first 5 letters of x: $x_0x_1x_2x_3x_4 = \mathtt{01010}$ and a contradiction with the overlap-freeness of τ. ■

Proposition 1.10. *In τ for every $k \in \mathbb{N}$: $\psi^k(xx)$ is located at position $2^{k+1}j + 2^k$, where $x \in \{\mathtt{0}, \mathtt{1}\}$ and $j \in \mathbb{N}$.*

Proof. Let's denote a sentence "In τ $\psi^k(xx)$ is located at position $2^{k+1}j + 2^k$, where $x \in \{\mathtt{0}, \mathtt{1}\}$ and $j \in \mathbb{N}$" by $T(k)$. We use the mathematical induction on k.

$T(k = 0)$: $\mathtt{00}$ and $\mathtt{11}$ are never fully contained in one of the segments $\mathtt{01}$ and $\mathtt{10}$, which implies that the first letters of these squares are always located at odd positions.

$T(k) \Rightarrow T(k+1)$: Let a word $u\psi^k(xx)$ be a subword of τ, where $u \in \{\mathtt{0}, \mathtt{1}\}^*$ and the length of u equals $2^{k+1}j + 2^k$.

$$\psi(u\psi^k(xx)) = \psi(u)\psi^{k+1}(xx)$$

is also a subword of τ, thanks to $\psi(\tau) = \tau$. The length of $\psi(u)$ is equal to $2 \cdot (2^{k+1}j + 2^k) = 2^{k+2}j + 2^{k+1}$, because every application of ψ doubles the length of words. Moreover, Proposition 1.8 implies that $\psi^{k+1}(xx)$ can only be created after applying ψ to $\psi^k(xx)$.

Therefore $T(k)$ is true for every $k \in \mathbb{N}$. ■

Proposition 1.11. *If we substitute any number of letters at odd positions by the third letter $\mathtt{2}$ in τ, we obtain a ternary cubefree sequence (but not overlap-free).*

Proof. Let's call τ' the sequence obtained from τ after substitutions. Assume that τ' has an odd cube $w = xxx$ of length k. Because of cubefreeness of τ, there exists an index j $(0 \leq j < k)$ that

$$x_j = w_j = w_{j+k} = w_{j+2k} = \mathtt{2}$$

which is a contradiction with the even distance between particular $\mathtt{2}$s.

For the even case, thanks to Proposition 1.7, we should notice that for every index i: $\tau_i = \tau_{2i} = \tau'_{2i}$. Assume that τ' has an even cube $w = xxx$ of length k. One of these two words:

$$w_0w_2 \cdots w_{k-2}w_k \cdots w_{2k-2}w_{2k} \cdots w_{3k-2} = (x_0x_2 \cdots x_{k-2})^3,$$
$$w_1w_3 \cdots w_{k-1}w_{k+1} \cdots w_{2k-1}w_{2k+1} \cdots w_{3k-1} = (x_1x_3 \cdots x_{k-1})^3$$

is a subword of τ. Anyway, we receive a contradiction with the cubefreeness of τ.

To complete the proof we show the sample of overlaps in the sequence τ'. From the definition of τ we know that

$$\psi^4(\mathtt{0}) = \mathtt{0}\underline{\mathtt{1}}\mathtt{1}\underline{\overline{\mathtt{0}}}\mathtt{1}\underline{\mathtt{0}}\mathtt{0}\overline{\mathtt{1}}\mathtt{100}\overline{\mathtt{1}}\mathtt{0110}$$

is its subword. If we put $\mathtt{2}$ on positions $1, 3, 5$ (respectively $3, 7, 11$) we obtain $\mathtt{21212}$ ($\mathtt{210021002}$) as an overlap. ∎

An infinite squarefree ternary word was also found by Thue [33], but the substitution below comes from [26]. The original construction of the squarefree word was more complicated.

Theorem 1.12. *Let's define a substitution* $\phi : \{\mathtt{a}, \mathtt{b}, \mathtt{c}\} \to \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}^*$*:*

$$\mathtt{a} \to \mathtt{abc}$$
$$\mathtt{b} \to \mathtt{ac}$$
$$\mathtt{c} \to \mathtt{b}$$

The sequence $t = \lim_{i\to\infty} \phi^i(\mathtt{a})$ *is squarefree.*

It is easy to notice that there is no squarefree binary sequence of length greater than 3. Let's look at other facts about words without squares.

Proposition 1.13. *Every squarefree word remains squarefree after insertions of a new letter between the symbols of the word, as long as no trivial squares are created.*

Proof. Assume that we obtain a square after the insertions. It doesn't contain only the new letters, because no trivial squares are created. It means that the original sequence has a square – a contradiction. ∎

Proposition 1.14. *If* w *is a subword of* t*, then* $||w|_{\mathtt{a}} - |w|_{\mathtt{c}}| \leq 1$.

Proof. Since t is a sequence built on the three words $\mathtt{abc}$, $\mathtt{ac}$ and $\mathtt{b}$, there is always a single $\mathtt{a}$ between any two consecutive $\mathtt{c}$'s in t. This implies that the difference between the number of these letters in any subword w of t cannot exceed 1. ∎

Proposition 1.15. *A letter* $\mathtt{b}$ *occurs only at odd positions of* t.

Proof. $t = \lim_{i\to\infty} \phi^i(\mathtt{a}) = \lim_{i\to\infty} (\phi^2)^i(\mathtt{a})$, where ϕ^2 can be written as:

$$\mathtt{a} \to \mathtt{abcacb}$$
$$\mathtt{b} \to \mathtt{abcb}$$
$$\mathtt{c} \to \mathtt{ac}$$

Each word on the right side is even and every occurrence of $\mathtt{b}$ inside them is located at odd position. Since t is built on these words, the proof is completed. ∎

Proposition 1.16. *There exist arbitrarily long sequences over a 3-letter alphabet such that no two subwords separated by exactly one symbol are identical.*

Proof. Consider a sequence s of length $2p$ obtained by doubling each term of t: $s_{2i} = s_{2i+1} = t_i$ for $0 \leq i < p$. We claim that the sequence s cannot contain subwords of the form bxb, where b is a nonempty sequence of length k, while x is a single symbol. To prove it suppose on the contrary that some subword of s has the form bxb, and let $x = s_j$, $j \geq k$. Then we have:

$$s_{j-k} \cdots s_{j-2}s_{j-1} = b_0 b_1 \cdots b_{k-1} = s_{j+1}s_{j+2} \cdots s_{j+k}$$

We may assume that j is even (the other case will follow by reversing the sequence s). It means that $s_{j+1} = s_{j-k} = x$. If $j - k$ is even, then also $s_{j-k+1} = x$, which in turn implies that $s_{j+2} = x = s_j$. Actually, in this case we get that all terms of b are equal to x, which clearly contradicts the squarefreeness of t. If $j-k$ is odd, then:

$$s_{j-k}s_{j-k+2} \cdots s_{j-1} = s_{j+1}s_{j+3} \cdots s_{j+k}$$

and we get a square of length $\frac{k+1}{2}$ in t – again a contradiction. ■

The following two theorems were proved by Entringer, Jackson and Schatz [17].

Theorem 1.17. *Let's define a substitution* $\sigma : \{\mathtt{a},\mathtt{b},\mathtt{c}\} \to \{\mathtt{0},\mathtt{1}\}^*$*:*

$$\begin{aligned} \mathtt{a} &\to \mathtt{1010} \\ \mathtt{b} &\to \mathtt{1100} \\ \mathtt{c} &\to \mathtt{0111} \end{aligned}$$

The sequence $g = \sigma(t) = \sigma(\lim_{i\to\infty} \phi^i(\mathtt{a}))$ *has only squares of length less than* 3.

Theorem 1.18. *Every binary sequence of length greater than* 18 *has squares of length* 2 *or greater.*

Although the bound of length above is optimal, it is possible to reduce the number of distinct squares. An infinite binary word with only 3 squares was revealed for the first time by Fraenkel and Simpson [18], but their solution was later simplified in [31] and [23]. The next substitution comes from [23].

Theorem 1.19. *Let's define a substitution* $\rho : \{\mathtt{a},\mathtt{b},\mathtt{c}\} \to \{\mathtt{0},\mathtt{1}\}^*$*:*

$$\begin{aligned} \mathtt{a} &\to \mathtt{111000110010110001110010} \\ \mathtt{b} &\to \mathtt{111000101100011100101100010} \\ \mathtt{c} &\to \mathtt{111000110010110001011100101100} \end{aligned}$$

The sequence $h = \rho(t) = \rho(\lim_{i\to\infty} \phi^i(\mathtt{a}))$ *has only three squares:* $\mathtt{00}$, $\mathtt{11}$ *and* $\mathtt{0101}$.

Every inifnite binary sequence with only three squares: $\mathtt{00}$, $\mathtt{11}$ and $\mathtt{0101}$ will be called a *3-square sequence*. We present some general properties of such sequences.

Proposition 1.20. *Every 3-square sequence doesn't have:*
(1) overlaps other than trivial cubes, (2) trivial 4*-th powers.*

Proof. (1) 3-square sequences have only one non-trivial square $\mathtt{0101}$, from which we may potentially create an overlap by adding $\mathtt{1}$ to the beginning or $\mathtt{0}$ to the end. However, both 10101 and 01010 contain $(\mathtt{10})^2$, which is a forbidden square.

(2) A trivial 4-th power is also a square of length 2 other than allowed $\mathtt{0101}$ in 3-square sequences. ∎

Proposition 1.21. *Every 3-square sequence has cubes:* $\mathtt{000}$ *or* $\mathtt{111}$.

Proof. Assume that there exist a 3-square sequence s without trivial cubes. From Theorem 1.18, s must contain an infinite number of a word $\mathtt{0101}$. Let's look at one of these subwords, but not at the first – in order to be sure that letters before the subword exist in the sequence. Suppose that $s_i s_{i+1} s_{i+2} s_{i+3} = \mathtt{0101}$ for a certain index i. $s_{i-1} = \mathtt{0}$ and $s_{i+4} = \mathtt{1}$, because of Proposition 1.20(1). Our aim is to avoid trivial cubes, so $s_{i-2} = \mathtt{1}$ and $s_{i+5} = \mathtt{0}$. Consider symbols $s_{i+6}s_{i+7}$ and the four cases dependent on their values:

$$\begin{aligned}
s_{i+6}s_{i+7} &= \mathtt{00} \Rightarrow s_{i+5}s_{i+6}s_{i+7} = \mathtt{0}^3 \\
s_{i+6}s_{i+7} &= \mathtt{01} \Rightarrow \text{no forbidden repetitions} \\
s_{i+6}s_{i+7} &= \mathtt{10} \Rightarrow s_{i+4}s_{i+5}s_{i+6}s_{i+7} = (\mathtt{10})^2 \\
s_{i+6}s_{i+7} &= \mathtt{11} \Rightarrow s_{i+2}s_{i+3}s_{i+4}s_{i+5}s_{i+6}s_{i+7} = (\mathtt{011})^2
\end{aligned}$$

Therefore, $s_{i+6}s_{i+7} = \mathtt{01}$. Similarly, we show that $s_{i-4}s_{i-3} = \mathtt{01}$. However:

$$s_{i-4}s_{i-3}s_{i-2}s_{i-1}s_i s_{i+1}s_{i+2}s_{i+3}s_{i+4}s_{i+5}s_{i+6}s_{i+7} = \mathtt{0110}\underline{\mathtt{0101}}\mathtt{1001} = (\mathtt{011001})^2$$

We receive a contradiction with the definition of s, which ends the proof. ∎

Proposition 1.22. *If we substitute every* $\mathtt{0101}$ *in any 3-square sequence by* $\mathtt{0201}$ *or* $\mathtt{0121}$*, where* $\mathtt{2}$ *is the third letter, we obtain a ternary sequence with only two squares:* $\mathtt{00}$ *and* $\mathtt{11}$.

Proof. Let s be a 3-square sequence and s' be s after the substitutions. Obviously, a square $\mathtt{22}$ is not a subword of s' – it would have to be a subword of $\mathtt{012201}$, which contradicts Proposition 1.20(1). There is no $\mathtt{0101}$ in s', so every potential non-trivial square in s' must contain $\mathtt{2}$. Assume that

$$xx = s'_i s'_{i+1} \cdots s'_{i+k-1} s'_{i+k} s'_{i+k+1} \cdots s'_{i+2k-1}$$

(for a certain index i and the length $k \geq 2$) and there exists j $(0 \leq j < k)$ such that $x_j = s'_{i+j} = s'_{i+k+j} = \mathtt{2}$. We consider two cases dependent on the letters s_{i+j} and s_{i+k+j}.

Suppose that $s_{i+j} \neq s_{i+k+j}$. Without loss of generality: $s_{i+j} = 0$ and $s_{i+k+j} = 1$. Thanks to the definition of the substitution we know that $s'_{i+j-1} = s'_{i+j+1} = 1$ and $s'_{i+k+j-1} = s'_{i+k+j+1} = 0$. If $j > 0$, then we obtain a contradiction from:

$$1 = s'_{i+j-1} = x_{j-1} = s'_{i+k+j-1} = 0$$

else $(j = 0)$ – from:

$$1 = s'_{i+j+1} = x_{j+1} = s'_{i+k+j+1} = 0$$

Let $s_{i+j} = s_{i+k+j}$. It implies that xx was a square before the substitutions, thus $xx =$ `0202` or $xx =$ `2121`. xx must be a subword of `020201` or `012121`, respectively. It contradicts Proposition 1.20(1). ■

Blind nonrepetitive games

2.1 Infinite partial words

A *partial word* over an alphabet A is a word over alphabet $A \cup \{\Diamond\}$, where $\Diamond$ is a letter not belonging to A, called a *hole*. A partial word might be *extended* to another partial word by filling some holes with letters from A. For instance, `ab◊acba◊cb◊c` is a partial word and `abcacba◊cbac` is one of its possible extensions.

We say that a partial word is a square (respectively: cube, overlap, mth power) if it can be extended to a non-partial word which is a square (cube, overlap, mth power). A non-partial word u is a subword of a partial word w if u is an extension of a certain partial subword of w. For example, `abc◊bc` is a square whereas `ccbc` and `bcb` are its subwords.

Our method is inspired by the results of Blanchet-Sadri et al. [8, 9, 10] concerning Thue properties of partial words, especially by the following theorem:

Theorem 2.1 (Blanchet-Sadri, Mercaş, Scott 2009). *There exists an infinite word over an* 8*-letter alphabet that remains non-trivial-squarefree after an arbitrary insertion of holes with the restriction that every two holes must have at least two non-hole symbols between them.*

The result above from [10] gives an explicit example of an infinite word w over 8 letters with the following, surprising property: if we make a partial word p from w by placing holes on arbitrary positions at distance at least 3 apart, then p cannot be extended to a word containing non-trivial squares. For instance, we may place holes on positions forming an arithmetic progression $0, 3, 6, 9, ...$ and the resulting partial word describes an explicit strategy for Ann in a *biased* version of the squarefree game, where she makes two moves in a row for one move of Ben.

A similar reasoning could be lead for the theorem from [9].

Theorem 2.2 (Blanchet-Sadri, Mercaş, Rashin, Willett 2012). *There exists an infinite word over a* 5*-letter alphabet that remains overlap-free after an arbitrary insertion of holes with the restriction that every two holes must have at least two non-hole symbols between them.*

After translation of the theorems above to the language of nonrepetitive games, we can observe that in both strategies Ann doesn't modify her future moves after Ben's turns, she just uses the letters from the sequence defined before the play starts. But what about the non-biased games? At first, let's state what kind of play we really mean:

Definition 2.3 (Blind nonrepetitive game)**.** *Fix a finite alphabet A and a positive integer n. The* squarefree game *of length n over A is played between two players, Ann and Ben. The players extend an initially empty word by alternately appending letters of A to its end. Ann knows when Ben makes a move, but she cannot see which letter he chooses. The game ends if the length of an emerging word has reached n, or if a predefined repetition of a certain kind (e.g. square, cube, overlap) has been created earlier. Ann wins if there are no such repetitions in the final word. Otherwise, Ben is the winner.*

Our considerations are lead for infinite partial words of a fixed appearance. For a sequence s, let's denote by $s_\Diamond$ the partial sequence $\Diamond s_0 \Diamond s_1 \Diamond s_2 \Diamond s_3 \cdots$ and name it: s with *holes shuffled*. Unfortunately, sequences with holes shuffled are generally neither non-trivial-squarefree, nor cubefree:

Proposition 2.4. *Every infinite sequence s with holes shuffled contains:*
(1) odd squares of any length, (2) trivial cubes.

Proof. (1) For every index i and length $k = 2j + 1$:

$$\Diamond s_i \Diamond s_{i+1} \cdots s_{i+j-1} \Diamond | s_{i+j} \Diamond s_{i+j+1} \Diamond \cdots \Diamond s_{i+k-1}$$

may be extended to

$$s_{i+j} s_i s_{i+j+1} s_{i+1} \cdots s_{i+j-1} s_{i+k-1} | s_{i+j} s_i s_{i+j+1} s_{i+1} \cdots s_{i+j-1} s_{i+k-1}$$

(2) For every index i: $\Diamond s_i \Diamond$ may be extended to $s_i s_i s_i$. ∎

Furthermore, in contrary to the biased version, the blind nonrepetitive game does not differentiate squares from overlaps.

Proposition 2.5. *Every infinite sequence s with holes shuffled is squarefree if and only if it is overlap-free.*

Proof. Obviously, if s with holes shuffled has an overlap, it contains also a square. Assume that $s_\Diamond$ has a subword xx. The length of xx is even, so either $\Diamond xx$, or $xx\Diamond$ is also a subword of $s_\Diamond$. Each of these two words is an overlap. ∎

2.2 Blind games outcomes

Let's see which sequences with holes shuffled define the winning strategies for Ann playing the blind games, remembering about unavoidable repetitions indicated by Proposition 2.4. For the games over a 3-letter alphabet we consider the squarefree word t from Theorem 1.12 and the overlap-free word τ from Theorem 1.6.

Proposition 2.6. *$t_\Diamond$ contains non-trivial cubes.*

Proof. From the definition of t we know that $\phi^2(\mathtt{a})$ and $\phi^4(\mathtt{a})$ are its subwords. $\phi^4(\mathtt{a}) = \phi^2(\mathtt{abcacb}) = \phi^2(\mathtt{ab})\phi^2(\mathtt{ca})\phi^2(\mathtt{cb})$, so $\phi^2(\mathtt{ca})$ is also a subword of t.

$$\phi^2(\mathtt{a})_\Diamond = (\mathtt{abcacb})_\Diamond = \underline{\Diamond\mathtt{a}\Diamond\mathtt{b}\Diamond\mathtt{c}\Diamond\mathtt{a}}\Diamond\mathtt{c}\Diamond\mathtt{b}$$
$$\phi^2(\mathtt{ca})_\Diamond = (\mathtt{acabcacb})_\Diamond = \underline{\Diamond\mathtt{a}\Diamond\mathtt{c}\Diamond\mathtt{a}\Diamond\mathtt{b}\Diamond\mathtt{c}\Diamond\mathtt{a}\Diamond\mathtt{c}\Diamond}\mathtt{b}$$

Thus $(\mathtt{bac})^3$ is a subword of $\phi^2(\mathtt{a})_\Diamond$ and $(\mathtt{aabcc})^3$ is a subword of $\phi^2(\mathtt{ca})_\Diamond$. ■

Proposition 2.7. *$t_\Diamond$ doesn't contain:*
(1) even squares, (2) cubes of length more than 5*, (3)* 4*th powers.*

Proof. (1) Assume that $t_\Diamond$ has an even square. The square must be an extension of $(\Diamond t_i \Diamond t_{i+1} \cdots \Diamond t_{i+k-1})^2$ or $(t_i \Diamond t_{i+1} \Diamond \cdots t_{i+k-1} \Diamond)^2$ for a certain index i and length k. However, it implies that $(t_i t_{i+1} \cdots t_{i+k-1})^2$ is a subword of t, which is a contradiction with the squarefreeness of t.

(2) Assume that $t_\Diamond$ has an odd cube of length $k \geq 7$. It must be an extension of

$$\Diamond t_i \Diamond t_{i+1} \cdots t_{i+j-1} \Diamond | t_{i+j} \Diamond t_{i+j+1} \Diamond \cdots \Diamond t_{i+k-1} | \Diamond t_i \Diamond t_{i+1} \cdots t_{i+j-1} \Diamond$$

or

$$t_i \Diamond t_{i+1} \cdots t_{i+j-1} \Diamond t_{i+j} | \Diamond t_{i+j+1} \Diamond \cdots \Diamond t_{i+k-1} \Diamond | t_i \Diamond t_{i+1} \cdots t_{i+j-1} \Diamond t_{i+j}$$

for a certain index i and length $k = 2j+1$. It implies that t contains two identical subwords $x = t_i t_{i+1} \cdots t_{i+j-1}$ with distance k between them. From Proposition 1.15 and the oddness of k we know that $|x|_{\mathtt{b}} = 0$. Thus $x = \mathtt{aca}$ or $x = \mathtt{cac}$, otherwise x would contain a square. However, in order to avoid the square the first letter after both subwords x must be $\mathtt{b}$, which contradicts Proposition 1.15.

(3) Assume that $t_\Diamond$ has an odd 4-th power, which must be an extension of

$$(\Diamond t_i \Diamond t_{i+1} \cdots t_{i+j-1} \Diamond | t_{i+j} \Diamond t_{i+j+1} \Diamond \cdots \Diamond t_{i+k-1})^2$$

or

$$(t_i \Diamond t_{i+1} \cdots t_{i+j-1} \Diamond t_{i+j} | \Diamond t_{i+j+1} \Diamond \cdots \Diamond t_{i+k-1} \Diamond)^2$$

for a certain index i and length $k = 2j+1$. However, it implies that $(t_i t_{i+1} \cdots t_{i+k-1})^2$ is a subword of t, which is a contradiction with the squarefreeness of t. ■

Definition 2.8. *Let's define an infinite sequence $\tau^{(2)}$ as:*

$$\tau_i^{(2)} = \begin{cases} \tau_i & \text{if } i \equiv 0 \pmod 2 \\ 2 & \text{if } i \equiv 1 \pmod 2 \end{cases}$$

Proposition 2.9. *$\tau_\Diamond^{(2)}$ doesn't contain: (1) non-trivial cubes, (2) trivial 4th powers.*

Proof. (1) Assume that $\tau_\Diamond^{(2)}$ has an even cube. We lead the reasoning like in the proof of Proposition 2.7(1) and use Proposition 1.11 to get a contradiction.

For the odd case we lead the similar reasoning to the one in the proof of Proposition 2.7(2). We obtain two identical subwords $x = \tau_i^{(2)}\tau_{i+1}^{(2)}\cdots\tau_{i+j-1}^{(2)}$ of t distant by $k = 2j+1, j \geq 1$ and a contradiction as a consequence of $|\tau_i^{(2)}\tau_{i+k}^{(2)}|_2 = 1$.

(2) For each i: $|\tau_i^{(2)}\tau_{i+1}^{(2)}|_2 = 1$, so $\Diamond\tau_i^{(2)}\Diamond\tau_{i+1}^{(2)}\Diamond$ doesn't have the 4th power. ■

We are ready to formulate corollaries related to the Ann's strategies for even, non-trivial and general blind games over a ternary alphabet.

Corollary 2.10. *There exists a strategy with finite description for Ann that allows her to win the blind even squarefree game of any length on 3 letters.*

Corollary 2.11. *There exists a strategy with finite description for Ann that allows her to win the blind non-trivial-cubefree game of any length on 3 letters.*

Corollary 2.12. *There exists a strategy with finite description for Ann that allows her to win the blind 4th-power-free game of any length on 3 letters.*

Proposition 2.4 implies the optimality of Corollaries 2.10, 2.11, 2.12 for a ternary and greater alphabets. It's time to switch to a binary alphabet and find winning strategies for Ann using τ from Theorem 1.6 and h from Theorem 1.19.

Proposition 2.13. *$\tau_\Diamond$ contains:*
(1) even squares of length not limited by any number,
(2) cubes of length 5 and 7, (3) 4th powers of length 3, (4) trivial 5th powers.

Proof. (1) It results directly from Proposition 1.9.

(2) From the definition of τ we know that $\psi^4(\mathtt{0})$ and $\psi^5(\mathtt{0})$ are its subwords. $\psi^5(\mathtt{0}) = \psi^2(\mathtt{01})\psi^2(\mathtt{101})\psi^2(\mathtt{001})$, so $\psi^2(\mathtt{101})$ is also a subword of τ.

$$\psi^2(\mathtt{101})_\Diamond = \Diamond 1 \Diamond 0 \underline{\Diamond 0 \Diamond 1 \Diamond 0 \Diamond 1 \Diamond 1 \Diamond 0 \Diamond 1 \Diamond} 0 \Diamond 0 \Diamond 1$$
$$\psi^4(\mathtt{0})_\Diamond = \Diamond 0 \Diamond 1 \Diamond 1 \underline{\Diamond 0 \Diamond 1 \Diamond 0 \Diamond 0 \Diamond 1 \Diamond 1 \Diamond 0 \Diamond 0 \Diamond 1 \Diamond 0 \Diamond 0 \Diamond} 1 \Diamond 1 \Diamond 0$$

Thus $(\mathtt{00111})^3$ is a subword of $\psi^2(\mathtt{101})_\Diamond$ and $(\mathtt{0011100})^3$ is a subword of $\psi^4(\mathtt{0})_\Diamond$.

(3) $(\mathtt{011})^4$ is also a subword of $\psi^2(\mathtt{101})_\Diamond$:

$$\psi^2(\mathtt{101})_\Diamond = \Diamond 1 \Diamond 0 \Diamond 0 \underline{\Diamond 1 \Diamond 0 \Diamond 1 \Diamond 1 \Diamond 0 \Diamond 1} \Diamond 0 \Diamond 0 \Diamond 1$$

(4) From the definition of τ we know that $\psi^2(\mathtt{0})$ is its subword and $\psi^2(\mathtt{0})_\Diamond$ contains 1^5, because $\psi^2(\mathtt{0})_\Diamond = \Diamond 0 \underline{\Diamond 1 \Diamond 1 \Diamond} 0$. ■

Proposition 2.14. $\tau_\Diamond$ *doesn't contain:*
(1) even cubes, (2) cubes of length more than 7*,*
(3) 4*th powers of length more than* 3*, (4) non-trivial* 5*th powers, (5)* 6*th powers.*

Proof. (1) The proof is similar to the one of Proposition 2.7(1).

(2) We apply the reasoning like in the proof of Proposition 2.7(2) and obtain two identical subwords $x = \tau_i\tau_{i+1}\cdots\tau_{i+j-1}$ of τ distant by $k = 2j+1, j \geq 4$. Because of Proposition 1.10 (k is not even), x does not have subwords `00` and `11`. Moreover, because τ is overlap-free, x does not contain subwords `10101` and `01010`. As a consequence, $x =$ `0101` or $x =$ `1010`. However, in order to avoid the overlap the first letter after both subwords x must be τ_{i+j-1}, so we get two words $\tau_{i+j-1}\tau_{i+j-1}$ distant by the odd value which contradicts Proposition 1.10.

(3) We lead the similar reasoning to the one in the proof of Proposition 2.7(3) and receive a contradiction with Proposition 1.9: τ doesn't have odd squares of length 5 or more.

(4) Assume that $\tau_\Diamond$ has an odd 5th power of length $k \geq 3$. It must extend:

$$(\Diamond\tau_i\Diamond\tau_{i+1}\cdots\tau_{i+j-1}\Diamond|\tau_{i+j}\Diamond\tau_{i+j+1}\Diamond\cdots\Diamond\tau_{i+k-1})^2\Diamond\tau_i\Diamond\tau_{i+1}\cdots\tau_{i+j-1}\Diamond$$

or

$$(\tau_i\Diamond\tau_{i+1}\cdots\tau_{i+j-1}\Diamond\tau_{i+j}|\Diamond\tau_{i+j+1}\Diamond\cdots\Diamond\tau_{i+k-1}\Diamond)^2\tau_i\Diamond\tau_{i+1}\cdots\tau_{i+j-1}\Diamond\tau_{i+j}$$

for a certain index i and length $k = 2j+1$. Thanks to $j \geq 1$, τ must contain a subword $(\tau_i\tau_{i+1}\cdots\tau_{i+k-1})^2\tau_i$. τ is overlap-free, so we get a contradiction.

(5) The proof is similar to the one of Proposition 2.7(3). ∎

In the propositions below we may use any 3-square sequence instead of h from Theorem 1.19.

Proposition 2.15. $h_\Diamond$ *contains:*
(1) squares of length 4*, (2) cubes of length* 3*, (3) trivial* 7*th powers.*

Proof. (1) h is a 3-square sequence, so $h_\Diamond$ contains $(\mathtt{1011})^2$ extending $\Diamond\mathtt{0}\Diamond\mathtt{1}\Diamond\mathtt{0}\Diamond\mathtt{1}$.

(2) h has a subword `01011`, see Proposition 1.20(1). As a consequence, $h_\Diamond$ contains $(\mathtt{011})^3$ extending $\Diamond\mathtt{1}\Diamond\mathtt{0}\Diamond\mathtt{1}\Diamond\mathtt{1}\Diamond$.

(3) From Proposition 1.21 we know that $h_\Diamond$ contains $\mathtt{0}^7$ or $\mathtt{1}^7$ extending $\Diamond\mathtt{0}\Diamond\mathtt{0}\Diamond\mathtt{0}\Diamond$ or $\Diamond\mathtt{1}\Diamond\mathtt{1}\Diamond\mathtt{1}\Diamond$, respectively. ∎

Proposition 2.16. $h_\Diamond$ *doesn't contain:*
(1) even squares of length more than 4*, (2) even cubes of length more than* 2*,*
(3) non-trivial 4*th powers, (4)* 8*th powers.*

Proof. (1) We lead the similar reasoning to the one in the proof of 2.7(1) and obtain a contradiction because h doesn't have squares of length more than 2.

(2) The proof as above and a contradiction because of Proposition 1.20(1).

(3) For odd 4-th powers the proof like in 2.7(3). The only even 4-th powers we should consider are of length 2, but they contradict Proposition 1.20(2).

(4) It is enough to show that $h_\Diamond$ doesn't have trivial 8-th powers. It is true because of Proposition 1.20(2). ■

As in the case of a ternary alphabet, we formulate corollaries for the three variants of blind games over a binary alphabet.

Corollary 2.17. *There exists a strategy with finite description for Ann that allows her to win the blind even cubefree game of any length on* 2 *letters.*

Corollary 2.18. *There exists a strategy with finite description for Ann that allows her to win the blind non-trivial-*4*th-power-free game of any length on* 2 *letters.*

Corollary 2.19. *There exists a strategy with finite description for Ann that allows her to win the blind* 6*th-power-free game of any length on* 2 *letters.*

Interestingly, Corollaries 2.17, 2.18 and 2.19 indicate for a 2-letter alphabet the optimal results, although it is not as obvious as in the case of the corollaries for a 3-letter one. We end the chapter with a proposition implying the mentioned fact.

Proposition 2.20. *Every infinite binary sequence s with holes shuffled contains: (1) even squares, (2) non-trivial cubes, (3)* 5*th powers.*

Proof. Firstly, let's fix the alphabet: $A = \{\mathtt{0}, \mathtt{1}\}$.

(1) The longest squarefree binary words are `010` and `101`, so every longer binary sequence with holes shuffled must contain an even square.

(2) Assume that $s_\Diamond$ has no non-trivial cubes, especially the ones of length 2, 3 and 4, so s should not contain subwords: $u = \alpha\alpha\alpha$, $v = \alpha\beta\gamma\alpha$ and $w = \alpha\beta\alpha\beta\alpha\beta$, respectively ($\alpha, \beta, \gamma \in A$). Without loss of generality: $s_0 = \mathtt{0}$. If $s_1 = \mathtt{0}$, the next elements of s are determined by avoiding u and v: $s_2 = \mathtt{1}$ (u), $s_3 = \mathtt{1}$ (v), $s_4 \neq \mathtt{0}$ (v) and $s_4 \neq \mathtt{1}$ (u) – a contradiction.

It means that $s_1 = \mathtt{1}$. If we choose $s_2 = \mathtt{1}$, we receive the similar contradiction to the latest one, so $s_2 = \mathtt{0}$. In order to avoid v the next three letters of s would be $s_3 = \mathtt{1}, s_4 = \mathtt{0}, s_5 = \mathtt{1}$, yet it contradicts w and finally negates the existence of $s_\Diamond$ without non-trivial cubes.

(3) Assume that $s_\Diamond$ has no 5th powers. The presence of subwords `00` and `11` in s results in trivial 5th-powers in $s_\Diamond$. Consequently, the first ten letters of s should be either `0101010101`, or `1010101010`, which are 5th powers of length 4 in $s_\Diamond$. We received the contradiction finishing the proof. ■

Squarefree game

3.1 The first result

The contents of the section were published in our paper [20] and are essential to further considerations in the chapter. At first, let's start by providing a more formal setting for our problem.

Definition 3.1 (Squarefree game)**.** *Fix a finite alphabet A and a positive integer n. The* squarefree game *of length n over A is played between two players, Ann and Ben. The players extend an initially empty word by alternately appending letters of A to its end. The game ends if the length of an emerging word has reached n, or if a non-trivial square has been created earlier. Ann wins if there are no non-trivial squares in the final word. Otherwise, Ben is the winner.*

We assume that Ben makes the first move, though this has a little impact on the result. We prove that Ann can always win against Ben over a 9-letter alphabet. A winning strategy for Ann splits into two parts accordingly to the parity of squares. The even case is implied by Corollary 2.10. The odd case is somewhat more complicated.

Lemma 3.2. *Ann can avoid non-trivial odd squares while playing the squarefree game over a 3-letter alphabet.*

Proof. Ann's strategy amounts to the following two rules that we will call *invariants*:

I1 Ann never repeats the letter placed by Ben in his latest move.

I2 Ann maintains one of the letters marked as her *favourite*. She keeps using the favourite letter in subsequent moves until it violates the first invariant. When this happens, she chooses the new favourite which is the one different from the latest two letters used by Ben.

A more formal description is presented as a pseudocode in Algorithm **R2NOA3** – Ann's strategy for avoiding **R**epetitions: squares (**2**nd powers) **N**on-trivial **O**dd on the game over an **A**lphabet containing **3** symbols: `0`, `1` and `2`.

Algorithm R2NOA3

```
f ← 0                                  ▷ Ann's favourite
y ← 1                                  ▷ Ben's latest move
loop
    x ← opponentMove()                 ▷ returns 0, 1, or 2
    if x = f then
        f ← 3 − x − y                  ▷ select new favourite
    end if
    makeMove(f)
    y ← x                              ▷ remember Ben's latest move
end loop
```

To prove the assertion, assume on the contrary that at some point in the game an odd square $b_j b_{j+1} \cdots b_{j+2m-1} = ww$ occurs, where m is odd, $m \geq 3$. Dependent on which player places the first letter of the square b_j, we consider two cases. If Ben places b_j, then from invariant I1 we have

$$b_j \neq b_{j+1}, \quad b_{j+2} \neq b_{j+3}, \quad \cdots, \quad b_{j+m+1} \neq b_{j+m+2}, \quad \cdots, \quad b_{j+2m-2} \neq b_{j+2m-1}$$

However, as $w_i = b_{j+i} = b_{j+m+i}$, for $0 \leq i \leq m-1$, we may express this property simply as $w_i \neq w_{i+1}$ for $i = 0 \ldots m-2$, and additionally we have $w_{m-1} \neq w_0$. Observe that for the whole time ww is created Ben never repeats Ann's latest letter and thus Ann's favourite letter remains the same. This means that

$$w_1 = w_3 = \cdots = w_{m-2} = w_0 = w_2 = \cdots = w_{m-1}$$

which contradicts inequality $w_{m-1} \neq w_0$.

Assume now that Ann places the letter b_j. Then invariant I1 forces that

$$b_{j+1} \neq b_{j+2}, \quad b_{j+3} \neq b_{j+4}, \quad \cdots, \quad b_{j+m} \neq b_{j+m+1}, \quad \cdots, \quad b_{j+2m-3} \neq b_{j+2m-2}$$

Similarly as above, we conclude that $w_i \neq w_{i+1}$ for $0 \leq i \leq m-2$, but this time w_{m-1} may be equal to w_0. Note that this is Ben who chooses the first letter of the second half of the square, namely $b_{j+m} = w_0$. He must repeat Ann's move, otherwise he could face a situation similar to the one described above. Repeating Ann's move forces her to change her favourite letter. This is the only favourite letter change that can occur during the construction of the whole ww. Consequently,

$$w_1 = w_3 = \cdots = w_{m-2}$$

as she places her new favourite letter at every other position in the second half of the square. She has chosen her new favourite carefully, thus $w_1 \neq w_0$, and, more importantly, $w_1 \neq w_{m-2}$. Again, we get a contradiction. The proof of the lemma is finished. ■

Theorem 3.3 (Grytczuk, Kosiński, Zmarz 2015). *There exists a strategy with finite description for Ann that allows her to win the squarefree game of any length on 9 letters.*

Proof. Let A be a 3-letter alphabet and let $A \times A$ be a 9-letter alphabet of pairs of letters form A. Ann's strategy is to play simultaneously on both coordinates accordingly to the strategies described in the proofs of Corollary 2.10 and Lemma 3.2. In this way she avoids even squares on the first coordinate and odd squares on the second coordinate. Clearly, both strategies have finite description, so the proof is complete. ■

3.2 The current result

The contents of the section were published in our paper [24], which is an amendment to [27]. The proof presented here is more transparent and compact.

Let's analyze Algorithm **R2NA8** – Ann's strategy for avoiding **R**epetitions: squares (**2**nd powers) **N**on-trivial on the game over an **A**lphabet containing **8** symbols, which are:

$$(\mathtt{0},\mathtt{a}),\ (\mathtt{0},\mathtt{b}),\ (\mathtt{0},\mathtt{c}),\ (\mathtt{1},\mathtt{a}),\ (\mathtt{1},\mathtt{b}),\ (\mathtt{1},\mathtt{c}),\ (\mathtt{2},\mathtt{d}),\ (\mathtt{3},\mathtt{d}) \tag{3.1}$$

We refer to symbols $(\mathtt{2},\mathtt{d})$ and $(\mathtt{3},\mathtt{d})$ as `d`-*characters*. We use the word t from Theorem 1.12 and store its letters in the array before executing the algorithm.

Lemma 3.4. *Algorithm* ***R2NA8*** *won't let Ann lose on any even square.*

Proof. Let's look at the second coordinates of (3.1). Ann either plays subsequent letters from the squarefree sequence t (line 23) or inserts `d` between them (line 21). We observe that in her next turn after placing a `d`-character, she always changes the favourite to some non-`d`-character (lines 7–12). From Proposition 1.13 Ann's sequence of moves is squarefree, so it is possible to adapt Proposition 2.7(1) and Corollary 2.10 to a 4-letter alphabet and end the proof. ■

We compare Algorithm **R2NA8** to Algorithm **R2NOA3** in the view of the first coordinates of (3.1). Firstly, Ann still never repeats Ben's moves. Secondly, in Algorithm **R2NOA3** letters `0`, `1` and `2` are used interchangeably, while in Algorithm **R2NA8** Ann plays `d`-characters `2` and `3` differently from `0` and `1`. Morevoer, we save the information about the latest `d`-character appended by Ben (lines 27–29) to bring it into play in a specific case (line 17).

Finally, in Algorithm **R2NA8** there are two possible situations (instead of one) in which Ann has to change her favourite letter:

1. Ben copies the latest Ann's non-`d`-character move (lines 13–19);

2. the current Ann's favourite character is a `d`-character (lines 7–12).

Algorithm R2NA8

```
1: f ← 0                                    ▷ Ann's favourite
2: y ← 2                                    ▷ Ben's latest move
3: z ← 2                                    ▷ Ben's latest d-character
4: count ← 0
5: loop
6:     x ← opponentMove()                   ▷ returns 0, 1, 2, or 3
7:     if isD(f) then                       ▷ select new favourite: it is a d-character
8:         if isD(x) then                   ▷ isD(a) = True if and only if a ∈ {2, 3}
9:             f ← 1 − y
10:        else
11:            f ← 1 − x
12:        end if
13:    else if x = f then                   ▷ select new favourite: Ben copies Ann's move
14:        if isD(y) then
15:            f ← 1 − x
16:        else
17:            f ← 5 − z                    ▷ Ann's favourite is a d-character now
18:        end if
19:    end if
20:    if isD(f) then                       ▷ play favourite with d if it is a d-character ...
21:        makeMove(f, d)
22:    else                                 ▷ ...or with the next letter from t otherwise
23:        makeMove(f, t[count])
24:        count ← count + 1
25:    end if
26:    y ← x                                ▷ remember Ben's latest move
27:    if isD(x) then
28:        z ← x                            ▷ remember Ben's latest d-character
29:    end if
30: end loop
```

The first case is similar to the one from Algorithm **R2NOA3**. If Ben repeats Ann's letter, she appends the symbol different from two latest Ben's moves. However, if both of his latest letters are non-d-characters, then she chooses the d-character other than recent Ben's one (line 17). In the second situation Ann always plays a non-d-character different from the latest one used by Ben.

Lemma 3.5. *Algorithm **R2NA8** won't let Ann lose on any non-trivial odd square.*

Proof. Assume on the contrary, that at some point in the game an odd square $b_j b_{j+1} \dots b_{j+2m-1} = ww$ occurs, where m is odd, $m \geq 3$. Dependent on which player places the first letter of the square b_j, we consider two cases. If Ben places b_j, then we still have, like in the proof of Lemma 3.2, $w_i \neq w_{i+1}$ for $0 \leq i \leq m-2$ and additionally $w_{m-1} \neq w_0$. Observe that Ben never repeats Ann's latest letter for the whole time ww is created, so either Ann's favourite letter remains the same, or it changes once because she places $b_{j+1} \in \{\mathtt{2}, \mathtt{3}\}$ as a consequence of Ben's choice of b_j equal to b_{j-1} (outside ww). In both cases we have $b_{j+3} = b_{j+5} = \cdots = b_{j+2m-1} \notin \{\mathtt{2}, \mathtt{3}\}$. Thus, since m is odd and greater than 1, the following contradiction occurs:

$$w_{m-1} \neq w_0 = b_{j+m} = b_{j+2m-1} = w_{m-1}$$

Assume now that Ann places the letter b_j. Similarly as above, we conclude that $w_i \neq w_{i+1}$ for $0 \leq i \leq m-2$, but this time w_{m-1} may be equal to w_0. Note that this is Ben who chooses the first letter of the second half of the square, namely $b_{j+m} = w_0$. If he doesn't repeat Ann's move, he could face a situation similar to the one described above: either Ann's favourite letter remains the same, or it changes once because she places $b_j = w_0 \in \{\mathtt{2}, \mathtt{3}\}$. Indeed, in the second case $b_{j+m} \in \{\mathtt{2}, \mathtt{3}\}$ and $b_{j+2} = b_{j+4} = \cdots = b_{j+m-1} \notin \{\mathtt{2}, \mathtt{3}\}$, so $b_{j+m} \neq b_{j+m-1}$. In both cases we may easily state a contradiction:

$$w_1 \neq w_2 = b_{j+2} = b_{j+m-1} = b_{j+m+1} = w_1$$

Thus Ben chooses b_{j+m} equal to b_{j+m-1} and forces b_{j+m+1} to be a d-character. We can observe two changes of Ann's favourite inside the square. Formally, her consecutive moves are:

$$b_j = b_{j+2} = \cdots = b_{j+m-1} = \alpha \notin \{\mathtt{2}, \mathtt{3}\},$$

$$b_{j+m+1} \in \{\mathtt{2}, \mathtt{3}\}, \quad b_{j+m+3} = \cdots = b_{j+2m-2} = \beta \notin \{\mathtt{2}, \mathtt{3}\}$$

Without loss of generality we assume $b_{j+m+1} = \mathtt{2}$. We notice that Ben cannot play d-characters at positions $b_{j+3}, b_{j+5}, \cdots, b_{j+m-2}$ (equal to β) and b_{j+m} (equal to α). It implies $b_{j+1} \neq \mathtt{2}$ because of line 17 ($m \geq 5$) or lines 13–15 ($m = 3$) in Algorithm **R2NA8**. Hence, we obtain a contradiction: $w_1 = b_{j+1} \neq b_{j+m+1} = w_1$ and end the proof. ∎

Theorem 3.6 (Kosiński, Mercaş, Nowotka 2018)**.** *There exists a strategy with finite description for Ann that allows her to win the squarefree game of any length on 8 letters.*

Proof. Lemmas 3.4 and 3.5 indicate Algorithm **R2NA8** as the wanted strategy. ∎

3.3 Allowing bigger squares

As we already know, the trivial squares are inevitable in the squarefree game. We may additionally relax the rules by allowing squares of a small even length. By using a binary word containing only squares of length not greater than 2 in place of the squarefree ternary word, we can quickly formulate a theorem:

Theorem 3.7. *There exists a strategy with finite description for Ann that allows her to win the squarefree game of any length on* 6 *letters, provided Ann doesn't lose on squares of length* 2 *and* 4*.*

Proof. We use Proposition 2.16(1) and Lemma 3.2 as in the proof of Theorem 3.3. Practically we use the binary word h from Theorem 1.19 (though in this case the other binary word g from Theorem 1.17 also matches) instead of the ternary word t from Theorem 1.12. ■

It's possible to eliminate squares of length 4, but it requires more effort. For this purpose we present Algorithm **R2N2A7** – Ann's strategy for avoiding **R**epetitions: squares (**2**nd powers) **N**on-trivial and not of length **2** on the game over an **A**lphabet containing **7** symbols, which are:

$$(\mathtt{0}, \mathtt{a}),\ (\mathtt{0}, \mathtt{b}),\ (\mathtt{1}, \mathtt{a}),\ (\mathtt{1}, \mathtt{b}),\ (\mathtt{2}, \mathtt{a}),\ (\mathtt{2}, \mathtt{b}),\ (\mathtt{3}, \mathtt{c}) \tag{3.2}$$

We use a word h' created from the word h from Theorem 1.19 as a result of changing each subword `0101` into `0201` (see Proposition 1.22) and applying a substitution: $\mathtt{0} \to \mathtt{a},\ \mathtt{1} \to \mathtt{b},\ \mathtt{2} \to \mathtt{c}$. We store subsequent letters of h' in the array before executing the algorithm.

Lemma 3.8. *Algorithm* ***R2N2A7*** *won't let Ann lose on any even square greater than* 2*.*

Proof. Let's look at the second coordinates of (3.2). Ann either plays subsequent letters from the sequence h' (lines 19, 22), or changes some subwords `acab` into `abcb` (lines 17, 13). From Proposition 1.22 Ann's sequence of moves contains only squares `aa` and `bb`, so it is possible to adapt the proof of Proposition 2.7(1) in order to receive a partial word with only even squares of length 2. ■

We analyze Algorithm **R2N2A7** in the view of the first coordinates of (3.2) and make some initial observations (mainly based on lines 1, 3, 8–10, 26–28):

O1 Ann never repeats the non-`3` letter placed by Ben in his latest move.

O2 Ann maintains one of letters `0`, `1` or `2` marked as her *favourite*. She keeps using the favourite letter in subsequent non-`3`-moves. If and only if Ben's latest symbol is equal to it, she chooses the new favourite which is the one different from the last two non-`3` letters appended by Ben. In the mentioned situation Ann's favourite changes even if she has to use `3` as her move.

O3 Ann never changes her favourite right after Ben places `3`.

If we concentrate only on letters `0`, `1` or `2`, invariants I1 and I2 related to Algorithm **R2NOA3** are the same as observations O1 and O2. Actually, the way Ann chooses

Algorithm R2N2A7

```
f ← 0                                   ▷ Ann's favourite, it's never 3
y ← 1                                   ▷ Ben's latest move
z ← 1                                   ▷ Ben's latest non-3-move
count ← 0
flag3c ← False                          ▷ True when Ann must play (3, c)
loop
    x ← opponentMove()                  ▷ returns 0, 1, 2, or 3
    if x = f then
        f ← 3 − x − z                   ▷ select new favourite
    end if
    if flag3c then
        flag3c ← False
        makeMove(3, c)                  ▷ change acab to abcb – phase 2
    else if h′[count] = c then
        if x = f and y = 3 then         ▷ avoid x3xx3x
            flag3c ← True
            makeMove(f, b)              ▷ change acab to abcb – phase 1
        else
            makeMove(3, c)              ▷ cover Ann's favourite with 3
        end if
    else
        makeMove(f, h′[count])
    end if
    count ← count + 1
    y ← x                               ▷ remember Ben's latest move
    if x ≠ 3 then
        z ← x                           ▷ remember Ben's latest non-3-move
    end if
end loop
```

her favourite is nearly the same, but occasionally she has to place a symbol 3 *covering* some of her favourites, what would be a chance of building a square for Ben.

Before we turn to more advanced properties, we need an extra definition: a subword u of a word w created together by Ann and Ben is at *Ann's position* (*Ben's position* respectively) if Ann (Ben) is the one who places u_0.

Lemma 3.9. *Every word generated by Algorithm **R2N2A7** and Ben's moves jointly doesn't contain the following subwords:*
(1) $\alpha\alpha$ *at Ben's position, where* $\alpha \in \{0, 1, 2\}$*;*
(2) $\alpha 3 \beta$ *at Ann's position, where* $\alpha, \beta \in \{0, 1, 2\}, \alpha \neq \beta$*;*
(3) $3\alpha 3$ *at Ann's position, where* $\alpha \in \{0, 1, 2, 3\}$*;*

(4) $\alpha 3\beta\alpha$ *at Ben's position, where* $\alpha \in \{0, 1, 2\}, \beta \in \{0, 1, 2, 3\}$*;*
(5) $\alpha 333\beta$ *at Ann's position, where* $\alpha, \beta \in \{0, 1, 2\}, \alpha \neq \beta$*;*
(6) $3\alpha\beta\gamma 3$ *at Ann's position, where* $\alpha, \beta, \gamma \in \{0, 1, 2, 3\}$*;*
(7) $\alpha\beta 33\beta\alpha$ *at Ben's position, where* $\alpha, \beta \in \{0, 1, 2\}$*.*

Proof. (1), (2) They are directly implied by O1 and O3 respectively.

(3) From (3.2) and the proof of Lemma 3.8, a subsequence of Ann's letters cc contradicts Proposition 1.22.

(4) According to O2 (for $\beta \neq 3$) and O3 (for $\beta = 3$), Ann's favourite does not change and is equal to α (different from β, thanks to (1)). It contradicts O2, because Ann should choose other letter after Ben's α.

(5) We use O3 twice.

(6) From (3), (3.2) and the proof of Lemma 3.8, a subsequence of Ann's moves can be either cac, or cbc, but it still contradicts Proposition 1.22: every c has to be surrounded by two as or by two bs, which means squares of length 2.

(7) $\alpha \neq \beta$, because of (1). Thanks to O3, we know that Ann's favourite is β in her first two moves, and then changes to α. It contradicts O2, since Ann's third letter should be different from two latest Ben's non-3-moves, and one of them is α. ■

Ater obtaining the results from Lemma 3.9, we can prove the absence of non-trivial odd squares in a quite compact way.

Lemma 3.10. *Algorithm **R2N2A7** won't let Ann lose on any square of length* 3*.*

Proof. Assume that at some point in the game a square $b_j b_{j+1} \cdots b_{j+5} = ww$ was constructed. For cases in which $b_{j+i} \neq 3$, $0 \leq i < 6$ the reasoning from the proof of Lemma 3.2 works as well. Let's consider every other possibility, dependent on the position of ww (numbers in parentheses are references to Lemma 3.9):

$w \in \{3\alpha\beta, \alpha 3\beta, \alpha\beta 3\}$, where $3 \neq \alpha \neq \beta \neq 3$. $ww = (3\alpha\beta)^2$ at Ben's position and $ww = (\alpha\beta 3)^2$ at Ann's position contradict (4), while the other four subcases contradict (2).

$w \in \{3\alpha\alpha, \alpha 3\alpha, \alpha\alpha 3\}$, where $\alpha \neq 3$. Except of $ww = (\alpha 3\alpha)^2$ at Ann's position, all subcases contradict (1).

$w \in \{33\alpha, 3\alpha 3, \alpha 33, 333\}$, where $\alpha \neq 3$. $ww = (33\alpha)^2$ at Ann's position and $ww = (\alpha 33)^2$ at Ben's position contradict (6), whereas the other six subcases contradict (3).

The only remaining case is $ww = (\alpha 3\alpha)^2$ at Ann's position. However, notice that the algorithm handles this situation at lines 14–17 and 11–13, which prevents Ann from appending $b_{j+4} = 3$ and commands her to play $b_{j+6} = 3$. Unfortunately for Ben: $b_{j+3} = \alpha \neq 3$, so he is unable to create a square of length 3 containing b_{j+6}. It ends the proof, because any other possibility of creating such square was already excluded. ■

Lemma 3.11. *Algorithm* ***R2N2A7*** *won't let Ann lose on any odd square of length at least* 5.

Proof. Assume on the contrary, that at some point in the game an odd square $b_j b_{j+1} \dots b_{j+2m-1} = ww$ occurs, where m is odd, $m \geq 5$. We introduce some auxilliary definitions related to the square. Ben's $\mathtt{3}$-move b_{j+i} $(0 \leq i < 2m)$ *can be interpreted* as a letter $\alpha \in \{\mathtt{0}, \mathtt{1}, \mathtt{2}\}$ if:

$$i \notin \{0, 1, m, m+1\} \text{ and } b_{j+i-2} = \alpha$$

or

$$i \notin \{m-2, m-1, 2m-2, 2m-1\} \text{ and } b_{j+i+2} = \alpha \neq b_{j+i+1}$$

It means that b_{j+i} may be the same as at least one of its neighbours appended by Ben inside the singular w. Notice that the precondition $\alpha \neq b_{j+i+1}$ in the second case is relevant: if we interpret b_{j+i} as $\alpha = b_{j+i+1} = b_{j+i+2}$, we obtain a contradiction with O1.

By the procedure of *uncovering* a subword u of w we mean changing each $\mathtt{3}$ inside u played by Ann in one w (at some b_i) into her hidden favourite and checking if Ben's move at the corresponding position in the other w (at b_{i+m} or b_{i-m}) can be interpreted as a letter equal to the favourite. If the procedure ends with success, we *uncover* the subword u.

Note that in uncovered u Ann has only one favourite per w. Assume without loss of generality that, according to O2, there exist $0 \leq i < m-2$: $b_{j+i} = b_{j+i+1} \neq b_{j+i+2}$ and b_{j+i}, b_{j+i+2} are Ann's letters. It implies that $b_{j+m+i} = b_{j+m+i+1}$ and $b_{j+m+i+1}$ is Ann's move repeating Ben's one – a contradiction with O1.

We will uncover two certain types of subwords of w in order to obtain almost pure non-$\mathtt{3}$ sequences so as to apply the reasoning similar to the one from the proof of Lemma 3.2 – though with more cases, because of some "threes" left uncovered.

$\alpha\mathtt{33}\beta$**,** where $\alpha, \beta \in \{\mathtt{0}, \mathtt{1}, \mathtt{2}\}, \alpha \neq \beta$. For the subword at Ann's position we directly apply the definition of uncovering for the second $\mathtt{3}$. We do the same for the subword at Ben's position and the first $\mathtt{3}$.

$\alpha\mathtt{3}\alpha$**,** where $\alpha \in \{\mathtt{0}, \mathtt{1}, \mathtt{2}\}$. If $\alpha\mathtt{3}\alpha \neq w_{m-3} w_{m-2} w_{m-1}$, then $\alpha\mathtt{3}\alpha\beta$ is a subword of w, because $m \geq 5$. From Lemma 3.9 (1) and (3) we have: $\alpha \neq \beta \in \{\mathtt{0}, \mathtt{1}, \mathtt{2}\}$, so we can easily apply the definition of uncovering. Otherwise, $\beta\alpha\mathtt{3}\alpha$ is a subword of w and we are able to repeat the reasoning. By the way, note that if $m = 3$, then $\alpha\mathtt{3}\alpha$ is impossible to be uncovered.

Morevoer, there are some types of words which cannot be subwords of w (numbers in parentheses are references to Lemma 3.9):

$\alpha\mathtt{3}\beta$**,** where $\alpha, \beta \in \{\mathtt{0}, \mathtt{1}, \mathtt{2}\}, \alpha \neq \beta$. At Ann's position it contradicts (2).

$\alpha 33\alpha$, where $\alpha \in \{0,1,2\}$. At Ben's position it contradicts (4).

333. At Ann's position it contradicts (3).

After the preceding considerations, we are ready to state a corollary:

$$\begin{gathered}\text{For each } i : (0 \le i < m \ \wedge\ b_{j+i} \text{ played by Ann}) \Rightarrow b_{j+i} \in \{\alpha, 3\}, \text{ and}\\ \text{for each } i : (m \le i < 2m \ \wedge\ b_{j+i} \text{ played by Ann}) \Rightarrow b_{j+i} \in \{\beta, 3\},\\ \text{where } \alpha, \beta \in \{0,1,2\}\end{gathered} \tag{3.3}$$

Additionally, after uncovering every $\alpha 3\alpha$ and $\alpha 33\beta$, only w_0, w_1, w_{m-2} or w_{m-1} may be equal to 3. For cases in which none of these symbols from w is 3 the reasoning from the proof of Lemma 3.2 works as well. Let's consider the remaining situations after the procedure of uncovering. We use already mentioned types of words, this time subwords of ww, but not w (one more time, numbers in parentheses are references to Lemma 3.9):

$\alpha 3\beta = b_{j+m-2+k}b_{j+m-1+k}b_{j+m+k}$ at Ben's position, where $\alpha, \beta \in \{0,1,2\}$, $k \in \{0,1\}$. Due to (3.3): $b_{j+m+k+1} = b_{j+2m-2+k} = w_{m-2+k} = b_{j+m-2+k} = \alpha$, so it contradicts (4).

$\alpha 3\alpha = b_{j+m-2+k}b_{j+m-1+k}b_{j+m+k}$ at Ann's position, where $\alpha \in \{0,1,2\}$, $k \in \{0,1\}$. Due to (3.3): $b_{j+m+k+1} = w_{k+1} = b_{j+k+1} = b_{j+m-2+k} = \alpha$, so it contradicts (1).

$\alpha 33\alpha = b_{j+m-3+k}b_{j+m-2+k}b_{j+m-1+k}b_{j+m+k}$ at Ann's position, where $\alpha \in \{0,1,2\}$, $k \in \{0,1,2\}$. Due to (3.3): $b_{j+m+k+1} = b_{j+2m-4+k} = w_{m-4+k} = b_{j+m-4+k}$, so it contradicts (7).

$\alpha 33\beta = b_{j+m-3+k}b_{j+m-2+k}b_{j+m-1+k}b_{j+m+k}$ at Ann's position, where $\alpha, \beta \in \{0,1,2\}$, $\alpha \neq \beta$, $k \in \{0,1,2\}$. Due to (3.3): $\beta = b_{j+m+k} = w_k = b_{j+k} = b_{j+m-3+k} = \alpha$, a contradiction.

$\alpha 33\beta = b_{j+m-3+k}b_{j+m-2+k}b_{j+m-1+k}b_{j+m+k}$ at Ben's position, where $\alpha, \beta \in \{0,1,2\}$, $\alpha \neq \beta$, $k \in \{0,1,2\}$. Due to (3.3): $\beta = b_{j+m+k} = b_{j+2m-3+k} = w_{m-3+k} = b_{j+m-3+k} = \alpha$, a contradiction.

$333 = b_{j+m-2+k}b_{j+m-1+k}b_{j+m+k}$ at Ben's position, where $k \in \{0,1\}$. Due to (3.3): $b_{j+m+k+1} = b_{j+2m-4+k} = w_{m-4+k} = b_{j+m-4+k} = \alpha$. If $b_{j+m-3+k} = \alpha$, it contradicts (1). Otherwise, it contradicts (5).

In order to finish the proof notice that 3333 cannot be a subword of ww, because it contradicts (3). ■

Theorem 3.12. *There exists a strategy with finite description for Ann that allows her to win the squarefree game of any length on 7 letters, provided Ann doesn't lose on squares of length 2.*

Proof. Lemmas 3.8, 3.10, and 3.11 imply that Algorithm **R2N2A7** is the wanted strategy. ■

Sparse overlap-free game

The contents of the chapter were published in our paper [20]. We start with a slightly more formal setting.

Definition 4.1 (Sparse overlap-free game)**.** *Let us fix a finite alphabet A and a positive integer n. The* sparse overlap-free game *of length n over A is played between two players, Ann and Ben. The players extend an initial partial word consisting of n holes by alternately filling the holes with letters from A (one hole per move). The game ends if all holes have been filled. Ann wins if there are no overlaps in the final word. Otherwise, Ben is the winner.*

We make two additional assumptions:

A1 Ben makes the first move,

A2 the length of the play $n = 2k$ is even.

Notice that changing the starting player is equivalent to changing parity of the game board. If the *second* player holds a winning strategy on a board of length n, they can use the strategy to win as the *first* player on board of length $n - 1$ (as Ann) or on board of length $n + 1$ (as Ben).

Ann's strategy looks as follows. Before the game starts she divides the board into k consecutive segments each of length two. Then she assigns the ith segment with the ith letter t_i of a squarefree sequence t from Theorem 1.12.

Assume now that the alphabet used in the game is $A = \{\mathtt{0}, \mathtt{1}, \mathtt{2}, \mathtt{3}\}$. To each of the letters $\mathtt{a}, \mathtt{b}, \mathtt{c}$ Ann assigns a set of four *special words* over A in the following way:

$$
\begin{aligned}
\mathtt{a} &\to \{\mathtt{01}, \mathtt{12}, \mathtt{23}, \mathtt{30}\} \\
\mathtt{b} &\to \{\mathtt{02}, \mathtt{13}, \mathtt{20}, \mathtt{31}\} \\
\mathtt{c} &\to \{\mathtt{03}, \mathtt{10}, \mathtt{21}, \mathtt{32}\}
\end{aligned}
\tag{4.1}
$$

Notice that the first letters of the special words form in each set the alphabet A, while the second letters of these words form three different cyclic permutations of the alphabet A. By this property Ann can apply the following simple strategy without ambiguity: whenever Ben makes his move in the ith segment, Ann responds by making a complementary move so that one of the special words corresponding to the letter t_i appears in this segment. For instance, if Ben puts `3` on the second position of a segment assigned with letter `b`, then Ann responds with `1`, since the unique special word matching with Ben's choice in this case is `13`. Notice that this strategy guarantees that when the game stops the resulting word is an image of the word t under the substitution defined by the three rules above involving special words.

Now we are going to prove that this is a winning strategy for Ann.

Theorem 4.2 (Grytczuk, Kosiński, Zmarz 2015). *There exists a winning strategy with finite description for Ann that allows her to win the sparse overlap-free game of any even length over a 4-letter alphabet (provided Ben starts the play).*

Proof. We will prove that the above strategy guarantees a win for Ann. Suppose on the contrary that an overlap $w = axaxa$ ($a \in A, x \in A^*$), appeared somewhere in a final word constructed during the game. We consider two cases dependent on the parity of m. If m is odd, then the partition of w into the segments determined by the initial partition of the board has one of the following two forms:

$$w = |ax_0|x_1x_2| \cdots |x_{m-2}x_{m-1}|ax_0|x_1x_2| \cdots |x_{m-2}x_{m-1}|a$$

or

$$w = a|x_0x_1| \cdots |x_{m-1}a|x_0x_1| \cdots |x_{m-1}a|$$

By Ann's strategy in every segment there is a special word, so the overlap w can be written as:

$$w = r_0r_1 \cdots r_{q-1}r_0r_1 \cdots r_{q-1}a$$

or

$$w = as_0s_1 \cdots s_{q-1}s_0s_1 \cdots s_{q-1}$$

where r_i and s_i are special words and $q = \frac{m+1}{2}$. But each special word occupying a given segment determines uniquely the letter of t assigned to this segment. Hence we get a square in the sequence t, which is a contradiction.

If m is even, then we also have two possible forms of w:

$$w = |ax_0|x_1x_2| \cdots |x_{m-1}a|x_0x_1| \cdots |x_{m-2}x_{m-1}|a = u'a$$

or

$$w = a|x_0x_1| \cdots |x_{m-2}x_{m-1}|ax_0|x_1x_2| \cdots |x_{m-1}a| = au'$$

Notice however that in both cases the word u' consists of exactly the same collection of special words that may be listed as:

$$ax_0,\ x_0x_1,\ x_1x_2,\ \cdots,\ x_{m-2}x_{m-1},\ x_{m-1}a \tag{4.2}$$

For each symbol $z \in A = \{\mathtt{0}, \mathtt{1}, \mathtt{2}, \mathtt{3}\}$ denote by $\overline{z}$ a corresponding element of the additive group $\mathbb{Z}_4$. We write the following equation, which is obviously true:

$$(\overline{x_0}-\overline{a})+(\overline{x_1}-\overline{x_0})+(\overline{x_2}-\overline{x_1})+\ldots+(\overline{x_{m-1}}-\overline{x_{m-2}})+(\overline{a}-\overline{x_{m-1}}) \equiv 0 \pmod 4 \quad (4.3)$$

By comparing (4.2) and (4.3) we notice a correspondence between special words and differences of elements of group $\mathbb{Z}_4$. From (4.1) we know that special words that are assigned to $\mathtt{a}$ correspond to a difference of 1 in (4.3), $\mathtt{b}$ to 2, $\mathtt{c}$ to 3, and there are no differences of value 0. Let u be a subword of t that was used to generate u'. We rewrite (4.3) as:

$$|u|_{\mathtt{a}} \cdot 1 + |u|_{\mathtt{b}} \cdot 2 + |u|_{\mathtt{c}} \cdot 3 \equiv 0 \pmod 4 \qquad (4.4)$$

From (4.4) we have $|u|_{\mathtt{a}} \equiv |u|_{\mathtt{c}} \pmod 2$ and we apply Proposition 1.14, which implies $|u|_{\mathtt{a}} = |u|_{\mathtt{c}}$. Recall that $|u|_{\mathtt{a}} + |u|_{\mathtt{b}} + |u|_{\mathtt{c}} = |u| = m+1$ is odd. A sum of added "ones" and "threes" is $0 \pmod 4$, while $|u|_{\mathtt{b}}$ (number of "twos") is odd. Hence, the total sum is congruent to $2 \pmod 4$, which contradicts (4.4). ■

The above result is optimal in the light of the following theorem. Ben can win the game that is played on a 3-letter alphabet in just 5 moves.

Theorem 4.3 (Grytczuk, Kosiński, Zmarz 2015). *Ann has no winning strategy on a 3-letter alphabet overlap-free game that starts with 10 or more holes.*

Proof. We will prove that Ann has no winning strategy in the sparse overlap-free game over a 3-letter alphabet that starts with 10 or more holes.

Assume that alphabet used in the game is $A = \{\mathtt{0}, \mathtt{1}, \mathtt{2}\}$. The board

$$x_0x_1x_2x_3x_4x_5x_6x_7x_8x_9$$

at the beginning of the game consists of 10 holes. Ben makes the first move by inserting $\mathtt{0}$ at position x_1. Ann cannot place $\mathtt{0}$ at position x_0 or x_2 because Ben subsequently will fill in an overlap $\mathtt{000}$ at positions $x_0x_1x_2$. If Ann puts any letter at position x_4 or further, then Ben will place the next $\mathtt{0}$ at x_2 on his turn. On her next turn Ann will be unable to prevent Ben from creating an overlap at positions $x_0x_1x_2$ or $x_1x_2x_3$. If Ann puts any letter at x_3, then Bob will double her letter at x_4, which will also result in an overlap, this time at positions $x_2x_3x_4$ or $x_3x_4x_5$.

In order not to end the game too early Ann has to insert a letter different from $\mathtt{0}$ at position x_0 or x_2. Without loss of generality Ann chooses $\mathtt{1}$. Consider the case in which Ann places the letter at x_0:

$$\mathtt{10}x_2x_3x_4x_5x_6x_7x_8x_9$$

Now Ben has a winning strategy – it is enough to fill all odd positions with $\mathtt{0}$ in the increasing order. On his next turn he places $\mathtt{0}$ at x_3. Ann has to place at x_2 a letter other than $\mathtt{0}$. Notice that the letter cannot be also $\mathtt{1}$, otherwise Ben will create an overlap $\mathtt{10101}$ at $x_0x_1x_2x_3x_4$. Thus Ann chooses $\mathtt{2}$ at x_2 and Ben responds with $\mathtt{0}$ at x_5. Again, Ann has only one possibility of move unless she wants the immediate loss. The same situation occurs one more time, so we obtain the board below:

$$\mathtt{10201020}x_8x_9$$

The last move belongs to Ben, who puts $\mathtt{1}$ at position x_8, creates an overlap and wins the game.

Consider the other case, in which Ann on the second turn places $\mathtt{1}$ at x_2:

$$x_0\mathtt{01}x_3x_4x_5x_6x_7x_8x_9$$

This time Ben needs to fill all even positions with $\mathtt{1}$ in the increasing order. We repeat the reasoning from the case above and finally obtain the following board:

$$x_0\mathtt{01210121}x_9$$

On his last turn Ben places $\mathtt{1}$ at x_0, which ends the game with his win and ends the proof as well. ∎

It is worth to notice that Ann can use the same strategy as in Theorem 4.2 to win the 4-letter non-sparse overlap-free game. Moreover, the assumptions A1 and A2 are not necessary: Ben can only append letters to the end of the augmenting word, so Ann can always react with a proper special word.

Corollary 4.4. *There exists a winning strategy with finite description for Ann that allows her to win the overlap-free game of any length over a* 4*-letter alphabet.*

We can also adapt the strategy for Ben over a 3-letter alphabet: the final words with overlaps from Theorem 4.3 have every second letter equal, so if he plays the same letter 5 times in a row, he wins.

Corollary 4.5. *Ann has no winning strategy on a* 3*-letter alphabet overlap-free game of length* 10 *or more.*

Game over a binary alphabet

5.1 5th-power-free game

We have already stated the squarefree game in Definition 3.1. Let's define a more general type of a nonrepetitive game in the same manner. Notice that this time Ann needs to avoid also trivial repetitions.

Definition 5.1 (mth-power-free game)**.** *Fix a finite alphabet A and positive integers n and m ($m \geq 3$). The* mth-power-free game *of length n over A is played between two players, Ann and Ben. The players extend an initially empty word by alternately appending letters of A to its end. The game ends if the length of an emerging word has reached n, or if an mth power has been created earlier. Ann wins if there are no mth powers in the final word. Otherwise, Ben is the winner.*

Let us think about the minimal m for which Ann has a winning strategy in the mth-power-free game on a binary alphabet. From Corollary 2.19 we know that $m \leq 6$ and from the theorem below: $m > 4$.

Theorem 5.2. *Ann has no winning strategy on a 2-letter alphabet 4th-power-free game of length 17 or more.*

Proof. Ben's tactic is to copy Ann's latest move: if Ann places the same letter twice in a row, he will easily obtain a trivial 4th power. It implies that Ann has to use `0` and `1` alternately, but after her eighth move Ben is able to create a 4th power of length 4, which means a 17th letter appended to the board if Ben starts the play: `00011001100110011`. ■

Trivial repetitions are so important that if Ann does not lose on trivial mth powers, m is decreased to at most 4 (Corollary 2.18). Let's consider an infinite binary overlap-free word τ from Theorem 1.6. From Propositions 2.13(4) and 2.14(4) the only 5th powers Ann would encounter playing the subsequent letters from τ are just the trivial ones. Proposition 2.20(3) implies that Ann must actively react for Ben's attempts to force a 5th power of length 1.

Algorithm R5A2

```
 1: y ← 1                                                   ▷ Ben's latest move
 2: count ← 0
 3: flagMoreChanges ← False        ▷ True if further modifications of tau possible
 4: loop
 5:     x ← opponentMove()                                  ▷ returns 0 or 1
 6:     if flagMoreChanges then
 7:         flagMoreChanges ← False
 8:         if x ≠ y then                                   ▷ False ⇒ (b1) or (c1)
 9:             tau[count]     ← 1 − tau[count]
10:             tau[count + 1] ← 1 − tau[count + 1]
11:             tau[count + 2] ← 1 − tau[count + 2]
12:             if x = tau[count + 3] then          ▷ True ⇒ (b2), False ⇒ (c2)
13:                 tau[count + 3] ← 1 − tau[count + 3]
14:                 tau[count + 4] ← 1 − tau[count + 4]
15:             end if
16:         end if
17:     else if tau[count] = x = tau[count − 1] = y then     ▷ let tau[−1] ≠ tau[0]
18:         if x ≠ tau[count + 2] then                      ▷ False ⇒ (a)
19:             flagMoreChanges ← True
20:         end if
21:         tau[count] ← 1 − tau[count]
22:     end if
23:     makeMove(tau[count])
24:     count ← count + 1
25:     y ← x                                   ▷ remember Ben's latest move
26: end loop
```

For that purpose we present Algorithm **R5A2** – Ann's strategy for avoiding **R**epetitions: **5**th powers on the game over an **A**lphabet containing **2** symbols: $\mathtt{0}, \mathtt{1}$. Before its execution we store subsequent letters of τ in the array *tau* which is modified by the algorithm during the runtime. An *output word* is the content of this array after Algorithm **R5A2** finishes the execution (see line 23). For simplicity we assume that every output word is infinite. The letter at position i in the output word might be different from τ_i.

As we know from the definition of ψ, τ is built on 2-letter segments. Notice that $\tau = \lim_{i\to\infty} \psi^i(\mathtt{0}) = \lim_{i\to\infty} (\psi^j)^i(\mathtt{0})$, where j is an arbitrary positive integer. As a consequence, we can divide τ into 2^j-letter segments, taken directly from the definition of ψ^j. In case of $j \in \{2, 3\}$ we get 4-letter segments and 8-letter segments of the form below, located in τ only at positions divisible by 4 and 8, respectively:

$$\mathtt{0110}, \quad \mathtt{1001} \quad \text{and} \quad \mathtt{01101001}, \quad \mathtt{10010110} \tag{5.1}$$

So as to describe Algorithm **R5A2** properly we introduce a notion for disturbance of τ. We will refer to a word u of length k as a *wave* when it is a subword of an output word w that $u_i = w_{j+i} \neq \tau_{j+i}$ and $w_{j+k} = \tau_{j+k}$, where $0 \leq i < k$ and j is the value of the variable *count* for which line 21 was executed.

Let's think about the possible waves. Consider the first one in an output word w: no symbol before this wave was changed by Algorithm **R5A2**. Let j be the value of *count* when the algorithm first time evaluates the condition from line 17 to `True`. Note that $j \geq 2$, which implies the existence of τ_{j-1}, τ_{j-2}. There is no trivial cube in τ, so $\tau_{j-2}\tau_{j-1}\tau_j\tau_{j+1} = abba$, where $(a, b) \in \{(\mathtt{0}, \mathtt{1}), (\mathtt{1}, \mathtt{0})\}$.

Let β_i mean a letter played by Ben between ith and $(i+1)$th Ann's moves. Notice that $\beta_{j-2} = \beta_{j-1} = b$. Now we are ready to distinguish five cases of Ann's behaviour dependent on letters from τ and Ben's move β_j:

(a) : $\tau_{j-2}\tau_{j-1}\tau_j\tau_{j+1}\tau_{j+2}\tau_{j+3} = abbaba$ (1 trivial square);

(b1) : $\tau_{j-2}\tau_{j-1}\tau_j\tau_{j+1}\tau_{j+2}\tau_{j+3}\tau_{j+4}\tau_{j+5} = abbaabab$ (2 trivial squares) and $\beta_j = b$;

(b2) : $\tau_{j-2}\tau_{j-1}\tau_j\tau_{j+1}\tau_{j+2}\tau_{j+3}\tau_{j+4}\tau_{j+5} = abbaabab$ (2 trivial squares) and $\beta_j = a$;

(c1) : $\tau_{j-2}\tau_{j-1}\tau_j\tau_{j+1}\tau_{j+2}\tau_{j+3}\tau_{j+4}\tau_{j+5} = abbaabba$ (3 trivial squares) and $\beta_j = b$;

(c2) : $\tau_{j-2}\tau_{j-1}\tau_j\tau_{j+1}\tau_{j+2}\tau_{j+3}\tau_{j+4}\tau_{j+5} = abbaabba$ (3 trivial squares) and $\beta_j = a$.

As we will see in the following lemma, these cases can be applied not only to the first wave, but also to every other one. Moreover, waves do not overlap (no wave starts inside another) and do not adjoin.

Lemma 5.3. *The only possible waves in output words are:*
waves of length 1 *in cases (a), (b1) and (c1);*
waves of length 4 *in case (c2);*
waves of length 6 *in case (b2).*
Furthermore, if there is a wave of length k at position j, then the next wave in the same output word (if exists) is located at position not less than $j + k + 2$.

Proof. The lengths provided above are obvious if line 17 of Algorithm **R5A2** is evaluated to `True` only for values i of *count* such that $tau[i] = \tau_i = tau[i-1] = \tau_{i-1}$.

Establish the first wave in an output word w located at position j. We consider situations in which:

$$\tau_{k-1} \neq w_{k-1} = w_k \quad \text{for} \quad k > j \tag{5.2}$$

It is a necessary condition to start the second wave inside or right after the first one, but it's not sufficient – we should check if $w_{k-1} = \beta_{k-1} = \beta_{k-2}$ additionally.

In cases (a), (b1) and (c1) we observe that (5.2) holds only for $k = j + 1$, but line 17 is evaluated to `False` because $\beta_{j-1} \neq w_j$. In case (c2) we observe that (5.2) holds only for $k = j + 2$, but line 17 is evaluated to `False` because $\beta_j \neq w_{j+1}$. Case (b2) is almost similar to (c2), but we must also notice that $\tau_{j+6} = \tau_{j+5} \neq w_{j+5}$;

otherwise $\tau_{j+2}\tau_{j+3}\tau_{j+4}\tau_{j+5}\tau_{j+6}$ would be an overlap. Thus $w_{j+6} = \tau_{j+6}$ and (5.2) does not hold for $k = j + 6$.

We already proved that if the first wave in w of length k is located at position j, then the second wave in w (if exists) is located at position not less than $j + k + 1$. We are able to expand this value to $j + k + 2$ by showing that the following equation is never true:

$$\tau_{j+k+1} = \beta_{j+k} = \tau_{j+k} = \beta_{j+k-1} \tag{5.3}$$

(5.3) is exactly line 17 adjusted to two letters from τ after the wave ($w_{j+k} = \tau_{j+k}$): we are about to decide whether w_{j+k+1} equals τ_{j+k+1} or not.

It is immediate to see that (5.3) does not hold for case (a): $\tau_{j+2} \neq \tau_{j+1}$; cases (b1) and (c1): $\tau_{j+2} = \tau_{j+1}$, but $\tau_{j+1} \neq \beta_j$; and case (c2): $\tau_{j+5} \neq \tau_{j+4}$. In case (b2) the inequality $\tau_{j+7} \neq \tau_{j+6}$ is the consequence of $\tau_{j+5}\tau_{j+6}\tau_{j+7}$ not being an overlap, which means (5.3) is false for all waves.

Knowing that after the first wave there are always at least two letters which don't belong to the second one, we are able to repeat the reasoning for the second, third and every other wave by using the mathematical induction. ■

We associate the cases with waves in order to use a term *(case)-wave*, where (case) is the one of (a), (b1), (b2), (c1) or (c2). A *(case)-wave gap* is a word $\tau_{j+k}\tau_{j+k+1}\cdots\tau_{j+p-1}$, in which k is the length of (case)-wave at position j in an output word w, and p is the smallest integer greater than k that $\tau_{j+p} \neq w_{j+p}$. If such p doesn't exist, we regard (case)-wave gap as an infinite subsequence of τ. A *minimal (case)-wave gap* means the (case)-wave gap of the minimum length among all positions of (case)-waves in all possible output words.

Finally, we define an *extended (case)-wave* at position j in an output word w as a catenation of $\tau_j\tau_{j+1}$, (case)-wave u that $u_0 = w_{j+2}$, and the minimal (case)-wave gap of u. (Notice that from Lemma 5.3: $w_jw_{j+1} = \tau_j\tau_{j+1}$). A word u is located at *$(f(k))$-positions* ($f : \mathbb{N} \to \mathbb{N}$) when there exist a nonempty subset K of $\mathbb{N}$ and an output word w that for every $k \in K$ the subword u is located at position $f(k)$ in w.

Lemma 5.4. *The only possible extended waves of output words are:*
extented (a)-waves `010010` *or* `101101` *located at* $(4k)$*-positions;*
extented (b1)-waves `01000101` *or* `10111010` *located at* $(8k+6)$*-positions;*
extended (b2)-waves `0101101010` *or* `1010010101` *located at* $(8k+6)$*-positions;*
extended (c1)-waves `010001` *or* `101110` *located at* $(8k+4)$*-positions;*
extended (c2)-waves `0101101010` *or* `1010010101` *located at* $(8k+4)$*-positions.*

Proof. Consider (case)-waves located at position j. Thanks to Lemma 5.3 we only have to find forms of $j-2$ and minimal (case)-wave gaps, which are always not shorter than 2. For the second purpose we search for the first positions after the current waves at which the next waves have a chance to appear in some output words.

Let's fix $(a, b) \in \{(\mathtt{0}, \mathtt{1}), (\mathtt{1}, \mathtt{0})\}$ and an output word w.

(a)-wave:

$$\begin{aligned} \tau_{j-2}\tau_{j-1}\tau_j\tau_{j+1}\tau_{j+2}\tau_{j+3} &= abbaba \\ w_{j-2}w_{j-1}w_jw_{j+1}w_{j+2} &= abaab \end{aligned}$$

$w_{j+2} \neq \tau_{j+3}$, so no wave is located at position $j+3$ and $w_{j+3} = \tau_{j+3}$. The absence of overlaps in τ implies $\tau_{j+4} = \tau_{j+3} = a$ and a possible wave at position $j+4$. After applying Proposition 1.10 to $\tau_j\tau_{j+1}\tau_{j+2}\tau_{j+3} = \psi(bb)$ we obtain $j = 4k+2$ and, consequently, $j-2 = 4k$.

(b1)-wave:

$$\begin{aligned} \tau_{j-2}\tau_{j-1}\tau_j\tau_{j+1}\tau_{j+2}\tau_{j+3}\tau_{j+4}\tau_{j+5}\tau_{j+6}\tau_{j+7} &= abbaababba \\ w_{j-2}w_{j-1}w_jw_{j+1}w_{j+2} &= abaaa \end{aligned}$$

By the same reasoning as above, we prove: $w_{j+3}w_{j+4}w_{j+5} = \tau_{j+3}\tau_{j+4}\tau_{j+5} = bab$ and the possibility of wave at position $j+6$. After applying Proposition 1.10 to $\tau_{j+2}\tau_{j+3}\tau_{j+4}\tau_{j+5} = \psi(aa)$ we obtain $j+2 = 4k+2$. Based on $j = 4k$ and (5.1), we observe that $\tau_j\tau_{j+1}\tau_{j+2}\tau_{j+3}$ and $\tau_{j-4}\tau_{j-3}\tau_{j-2}\tau_{j-1}$ are 4-letter blocks equal to $\psi^2(b)$. Once more we use Proposition 1.10 to get $j-4 = 8k+4$, which means $j-2 = 8k+6$.

(b2)-wave:

$$\begin{aligned} \tau_{j-2}\tau_{j-1}\tau_j\tau_{j+1}\tau_{j+2}\tau_{j+3}\tau_{j+4}\tau_{j+5}\tau_{j+6}\tau_{j+7} &= abbaababba \\ w_{j-2}w_{j-1}w_jw_{j+1}w_{j+2}w_{j+3}w_{j+4}w_{j+5}w_{j+6}w_{j+7} &= ababbababa \end{aligned}$$

Both (b1)-wave and (b2)-wave share the same subword of τ, so we can continue dividing it into blocks, this time 8-letter ones. Based on $j = 8k$ and (5.1), $\tau_j\tau_{j+1}\cdots\tau_{j+7}$ and $\tau_{j-8}\tau_{j-7}\cdots\tau_{j-1}$ are different 8-letter blocks. Thus we are unable to determine the next block and its first letter τ_{j+8} – it could be either equal to τ_{j+7}, or not. As a consequence, a wave at position $j+8$ is possible.

(c1)-wave:

$$\begin{aligned} \tau_{j-2}\tau_{j-1}\tau_j\tau_{j+1}\tau_{j+2}\tau_{j+3}\tau_{j+4}\tau_{j+5} &= abbaabba \\ w_{j-2}w_{j-1}w_jw_{j+1}w_{j+2} &= abaaa \end{aligned}$$

By the same reasoning as in the case of (a)-wave, we receive $w_{j+3} = \tau_{j+3} = b$ and the possibility of wave at position $j+4$. After applying Proposition 1.10 to $\tau_{j-2}\tau_{j-1}\cdots\tau_{j+5} = \psi^2(aa)$ we obtain $j-2 = 8k+4$.

(c2)-wave:

$$\begin{aligned} \tau_{j-2}\tau_{j-1}\tau_j\tau_{j+1}\tau_{j+2}\tau_{j+3}\tau_{j+4}\tau_{j+5} &= abbaabba \\ w_{j-2}w_{j-1}w_jw_{j+1}w_{j+2}w_{j+3}w_{j+4}w_{j+5} &= ababbaba \end{aligned}$$

Based on $j+2 = 8k$ and (5.1), $\tau_{j+2}\tau_{j+3}\cdots\tau_{j+9} = abbabaab$. By the same reasoning as in the case of (a)-wave, we receive $w_{j+6}w_{j+7} = \tau_{j+6}\tau_{j+7} = ba$ and the possibility of wave at position $j+8$. ■

A	B	A	B	A	B	A	B	A	B	A	B	A	B	A	B
0	1	1	1	$\boxed{0}$		0		1		0					
0		1		1		0		1		0		0		1	
1	0	0	0	$\boxed{1}$		1		0		1					
1		0		0		1		0		1		1		0	

Table 5.1: Extended (a)-waves with their waves and 4-letter segments of τ marked.

A	B	A	B	A	B	A	B	A	B	A	B	A	B	A	B	A	B	A	B
0	1	1	1	$\boxed{0}$	$\underline{1}$	0		0		1		0		1					
0		1		1		0		0		1		0		1		1		0	
1	0	0	0	$\boxed{1}$	$\underline{0}$	1		1		0		1		0					
1		0		0		1		1		0		1		0		0		1	

Table 5.2: Extended (b1)-waves with their waves and 4-letter segments of τ marked.

A	B	A	B	A	B	A	B	A	B	A	B	A	B	A	B	A	B	A	B
0	1	1	1	0	$\underline{0}$	1		1		0		1		0		1		0	
0		1		1		0		0		1		0		1		1		0	
1	0	0	0	1	$\underline{1}$	0		0		1		0		1		0		1	
1		0		0		1		1		0		1		0		0		1	

Table 5.3: Extended (b2)-waves with their waves and 4-letter segments of τ marked.

A	B	A	B	A	B	A	B	A	B	A	B	A	B	A	B
0	1	1	1	$\boxed{0}$	$\underline{1}$	0		0		1					
0		1		1		0		0		1		1		0	
1	0	0	0	$\boxed{1}$	$\underline{0}$	1		1		0					
1		0		0		1		1		0		0		1	

Table 5.4: Extended (c1)-waves with their waves and 4-letter segments of τ marked.

A	B	A	B	A	B	A	B	A	B	A	B	A	B	A	B	A	B	A	B	A	B	A	B
0	1	1	1	0	$\underline{0}$	1		1		0		1		0		1		0					
0		1		1		0		0		1		1		0		1		0		0		1	
1	0	0	0	1	$\underline{1}$	0		0		1		0		1		0		1					
1		0		0		1		1		0		0		1		0		1		1		0	

Table 5.5: Extended (c2)-waves with their waves and 4-letter segments of τ marked.

The graphic summary of Lemma 5.4 is presented in Tables 5.1–5.5. After comparing the contents of output words to the 4-letter segments from (5.1), we are able to formulate the following remarks to Lemma 5.4.

Remark 5.5. *Inside output words we notice subwords:*
`001` *and* `110` *only at* $(2k+1)$*-positions and* $(4k+2)$*-positions;*
`000` *and* `111` *only at* $(8k)$*-positions and* $(8k+6)$*-positions.*

Remark 5.6. *The differences in* 4*-letter segments between output words and* τ *are:*
(a)-waves and (c1)-waves at $(4k)$*-positions introduce*
`0100` *instead of* `0110` *and* `1011` *instead of* `1001`*;*
(b1)-waves at $(8k)$*-positions introduce*
`0001` *instead of* `1001` *and* `1110` *instead of* `0110`*;*
(b2)-waves at $(8k)$*-positions introduce consecutive segments*
`0110, 1010` *instead of* `1001, 0110` *and* `1001, 0101` *instead of* `0110, 1001`*;*
(c2)-waves at $(8k+4)$*-positions introduce consecutive segments*
`0101, 1010` *instead of* `0110, 0110` *and* `1010, 0101` *instead of* `1001, 1001`*.*

5.2 The proof

At the beginning of the section let's show the main difference between the effects of Algorithm **R5A2** and the blind tactic for Ann.

Lemma 5.7. *Algorithm* ***R5A2*** *won't let Ann lose on a* 5*th power of length* 1*.*

Proof. The definition of wave and Lemma 5.3 implies that for every output word w with a sequence β and for every index j: $\beta_j w_{j+1}\beta_{j+1}w_{j+2}$ is not a trivial 4th power. Thus $w_j\beta_j w_{j+1}\beta_{j+1}w_{j+2}$ and $\beta_j w_{j+1}\beta_{j+1}w_{j+2}\beta_{j+2}$ cannot be 5th powers. ■

After we got rid of trivial 5th powers we must check if the algorithm introduce some 5th powers of greater length, which $\tau_\Diamond$ doesn't contain. In each succeeding lemma we will assume there are an output word w with a corresponding sequence β, an index j and a word s of length p such that: if p is odd at least one of these two equations holds:

$$w_j\beta_j w_{j+1}\beta_{j+1}\cdots\beta_{j+\lfloor q\rfloor}w_{j+\lceil q\rceil} = s^5 \tag{5.4}$$
$$\beta_{j-1}w_j\beta_j w_{j+1}\cdots w_{j+\lfloor q\rfloor}\beta_{j+\lfloor q\rfloor} = s^5 \tag{5.5}$$

or if p is even at least one of these two equations holds:

$$w_j\beta_j w_{j+1}\beta_{j+1}\cdots w_{j+q}\beta_{j+q} = s^5 \tag{5.6}$$
$$\beta_{j-1}w_j\beta_j w_{j+1}\cdots\beta_{j+q-1}w_{j+q} = s^5 \tag{5.7}$$

where $q = \frac{5p-2}{2}$. We will sometimes use an additional integer q', which equals $\lceil q\rceil$ for (5.4) or $\lfloor q\rfloor$ for (5.5). Let's fix $(a,b)\in\{(\mathtt{0},\mathtt{1}),(\mathtt{1},\mathtt{0})\}$. We will lead our reasoning to contradictions usually by considering two cases for 5th-power candidates s^5:

C1 there exists an extended wave whose at least first three letters (so at least first one of its wave) belong to the candidate;

C2 there exists an extended wave which does not fulfil C1, but whose at least last letter of its wave belongs to the candidate.

Words which do not meet any of the critieria above are not considered as candidates anymore: they are just subwords of $\tau_{\Diamond}$. We observe that C1 can be expressed as the existence of an index r that:

$$\begin{aligned} w_{j+r}\beta_{j+r}w_{j+r+1}\beta_{j+r+1}w_{j+r+2} &= abbba \\ \tau_{j+r}\tau_{j+r+1}\tau_{j+r+2} &= abb \end{aligned} \tag{5.8}$$

where $j+r$ is even ($w_{j+r}w_{j+r+1}$ has to be a 2-letter segment) and $0 \le r \le q'-2$. If C1 holds, then for $p = 2k+1$ ($k \ge 2$) we can find integers $i, \hat{i}, \tilde{i}, \breve{i}$ that the following words are subwords of s^5:

$$\begin{aligned} \beta_{2i-k}w_{2i-k+1}\cdots\beta_{2i-1}w_{2i}\beta_{2i}w_{2i+1} &= ba\xi abbb \\ w_{2\hat{i}-k+2}\beta_{2\hat{i}-k+2}\cdots w_{2\hat{i}+1}\beta_{2\hat{i}+1}w_{2\hat{i}+2}\beta_{2\hat{i}+2} &= ba\xi abbb \\ \beta_{2\tilde{i}-k+1}w_{2\tilde{i}-k+2}\cdots\beta_{2\tilde{i}}w_{2\tilde{i}+1}\beta_{2\tilde{i}+1}w_{2\tilde{i}+2} &= ba\xi abbb \\ w_{2\breve{i}-k+1}\beta_{2\breve{i}-k+1}\cdots w_{2\breve{i}}\beta_{2\breve{i}}w_{2\breve{i}+1}\beta_{2\breve{i}+1} &= ba\xi abbb \end{aligned} \tag{5.9}$$

where ξ is some word of length $2k-4$ and

$$\begin{array}{rcl rcl} j+q'-\lceil q\rceil+k \le & 2i & \le j+q'-1, & j+k-2 \le & 2\hat{i} & \le j+\lfloor q\rfloor-2, \\ j+q'-\lceil q\rceil+k-1 \le & 2\tilde{i} & \le j+q'-2, & j+k-1 \le & 2\breve{i} & \le j+\lfloor q\rfloor-1 \end{array}$$

After matching the content of (5.9) to s^5 with r from (5.8) as the reference and small computations, we are able to determine the values of $i, \hat{i}, \tilde{i}, \breve{i}$. If k is odd, they are:

$$\begin{aligned} &i \in \{\tfrac{j+r-3k-1}{2}, \tfrac{j+r+k+1}{2}\}, && \hat{i} \in \{\tfrac{j+r-2k-2}{2}, \tfrac{j+r+2k}{2}\}, \\ &\tilde{i} \in \{\tfrac{j+r-k-1}{2}, \tfrac{j+r+3k+1}{2}\}, && \breve{i} \in \{\tfrac{j+r}{2}, \tfrac{j+r+4k+2}{2}\} \end{aligned} \tag{5.10}$$

Otherwise, these values are:

$$\begin{aligned} &i \in \{\tfrac{j+r-k}{2}, \tfrac{j+r+3k+2}{2}\}, && \hat{i} \in \{\tfrac{j+r-2k-2}{2}, \tfrac{j+r+2k}{2}\}, \\ &\tilde{i} \in \{\tfrac{j+r-3k-2}{2}, \tfrac{j+r+k}{2}\}, && \breve{i} \in \{\tfrac{j+r}{2}, \tfrac{j+r+4k+2}{2}\} \end{aligned} \tag{5.11}$$

Notice also that C2 implies the only one (case)-wave interfering with the candidate, therefore, for brevity, we can regard its (case)-wave gap as infinite. We will use Lemma 5.4 implicitly every time we mention properties of extended (case)-waves. In the succeeding lemmas concerning odd 5th powers the proofs are very similar, so we will expose the majority of technical details differentiating each reasoning.

Lemma 5.8. *Algorithm* ***R5A2*** *won't let Ann lose on a* 5*th power of length* 3.

Proof. Assume we have the word s of length 3 from (5.4) or (5.5). C1 implies $abbba$ is a subword of $(s_0s_1s_2)^5$, which is impossible due to $a \neq b$. We just need to verify C2. Observe that in such situation $w_j \cdots w_{j+6}$ is an overlap. Because τ is overlap-free, at least one letter from the overlap belongs to the wave interfering with s^5.

If we have an extended wave u with such interfering wave of length 1, then u_2 is the only letter if the wave and $u_2 = w_i$, where $i = j$ or $i = j+1$ (greater values of i would imply C1). (b1)-wave and (c1)-wave contradicts $w_j \cdots w_{j+6}$ being overlap, because $w_i = u_2 \neq u_5 = w_{i+3}$ and $i+3 \leq j+4 < j+6$.

For (a)-wave let $u_2 = a$. As a consequence, $\beta_{i-1} = b$ and $w_{i+1} = u_3 = u_2 = a$, so we obtain a word $ba\beta_i a$. When (5.5) holds or $i = j+1$, this word is a subword of $(s_0s_1s_2)^5$ – a contradiction. Otherwise $i = j$, $s_0 = u_2 = w_j$, and β_{i-1} doesn't belong to s^5. However, in (5.4) a word $w_{j+1} \cdots w_{j+7}$ must also be an overlap, but it contains only consecutive signs from τ – we get another contradiction.

Consider extended (b2)-wave: $u = ababbababa$. C2 means we should search for the overlap $w_j \cdots w_{j+6}$ that $i+1 \leq j \leq i+7$, where $w_i = u_0$. The infinite (b2)-wave gap means the equality $w_{i+8}w_{i+9}w_{i+10}\cdots = \tau_{i+8}\tau_{i+9}\tau_{i+10}\cdots$.

For $j \neq i+7$ we don't have overlaps, because:

$$w_{i+4} = u_4 = b \neq a = u_7 = w_{i+7}$$
$$w_{i+6} = u_6 = b \neq a = \tau_{i+9} = w_{i+9}$$

For $j = i+7$ in order to prove overlap-freeness we should know which 4-letter segment from (5.1) is $\tau_{i+10} \cdots \tau_{i+13}$:

$$w_{i+9} = \tau_{i+9} = a \neq b = \tau_{i+12} = w_{i+12} \quad (\tau_{i+10} \cdots \tau_{i+13} = abba)$$
$$w_{i+8} = \tau_{i+8} = b \neq a = \tau_{i+11} = w_{i+11} \quad (\tau_{i+10} \cdots \tau_{i+13} = baab)$$

At last, consider extended (c2)-wave: $u = ababbababa$. Similarly, $i+1 \leq j \leq i+5$ (where $w_i = u_0$) and $w_{i+6}w_{i+7}w_{i+8}\cdots = \tau_{i+6}\tau_{i+7}\tau_{i+8}\cdots$. We receive:

$$w_{i+4} = u_4 = b \neq a = \tau_{i+7} = w_{i+7}$$
$$w_{i+6} = \tau_{i+6} = b \neq a = \tau_{i+9} = w_{i+9}$$

Thus there is no overlap caused by (c2)-wave, which ends the proof. ■

Lemma 5.9. *Algorithm* ***R5A2*** *won't let Ann lose on a* 5*th power of length* 5.

Proof. Assume we have the word s of length 5 from (5.4) or (5.5). In C1 case note that (5.9) for $k = 2$ means 2-letter segment $w_{2i}w_{2i+1} = bb$ is not a part of τ and it could only be an effect of some 1-letter wave. Such wave should be preceded by $\beta_{2i-2}w_{2i-1}\beta_{2i-1} = aaa$, but $\beta_{2i-2} = b$ contradicts this statement.

Look at the remaining case C2. At first, we should exclude 1-letter waves. Observe we can find such waves at positions j or $j+1$, whereas $w_{j+2}w_{j+3}\cdots w_{j+11} = \tau_{j+2}\tau_{j+3}\cdots\tau_{j+11}$ is a square of length 5, which contradicts Proposition 1.9.

Now we examine an extended (b2)-wave or an extended (c2)-wave $u = ababbababa$ in the same way as in Lemma 5.8, yet this time we look for a square $w_j w_{j+1}\cdots w_{j+9}$ that $i+1 \le j \le i+7$ or $i+1 \le j \le i+5$ respectively, where $w_i = u_0$.

For the extended (b2)-wave (with usage of 8-letter segments from (5.1)):

$$
\begin{aligned}
w_{i+4} = u_4 = b \neq a = \tau_{i+9} = w_{i+9} & \\
w_{i+9} = \tau_{i+9} = a \neq b = \tau_{i+14} = w_{i+14} & \quad (\tau_{i+10}\cdots\tau_{i+17} = abbabaab) \\
w_{i+6} = u_6 = b \neq a = \tau_{i+11} = w_{i+11} & \quad (\tau_{i+10}\cdots\tau_{i+17} = baababba) \\
w_{i+11} = \tau_{i+11} = a \neq b = \tau_{i+16} = w_{i+16} & \quad (\tau_{i+10}\cdots\tau_{i+17} = baababba)
\end{aligned}
$$

For the extended (c2)-wave (with 8-letter segments as well):

$$
\begin{aligned}
w_{i+4} = u_4 = b \neq a = \tau_{i+9} = w_{i+9} & \\
w_{i+9} = \tau_{i+9} = a \neq b = \tau_{i+14} = w_{i+14} & \quad (\tau_{i+12}\cdots\tau_{i+19} = abbabaab) \\
w_{i+8} = \tau_{i+8} = b \neq a = \tau_{i+13} = w_{i+13} & \quad (\tau_{i+12}\cdots\tau_{i+19} = baababba)
\end{aligned}
$$

Thus we receive no square of length 5 and C2 doesn't hold. ■

Lemma 5.10. *Algorithm **R5A2** won't let Ann lose on a 5th power of length 7.*

Proof. Assume we have the word s of length 7 from (5.4) or (5.5). Let's consider C1 case. We apply (5.9) for $k=3$, which means a 2-letter segment $w_{2i}w_{2i+1} = bb$ is not a part of τ and it could only be an effect of some 1-letter wave. It implies $\xi = aa$ and $w_{2i-2}w_{2i-1}w_{2i} = aab$, which is not a prefix of any extended wave – a contradiction.

For C2 we make the steps as in Lemma 5.9. We apply Proposition 1.9 (squares of length 7) to eliminate 1-letter-wave subcases. Let's fix $w_i = u_0$, where $u = ababbababa$ is an extended (b2)-wave or an extended (c2)-wave, and search for a 7-letter square. We are unable to find it for the extended (b2)-wave:

$$
\begin{aligned}
w_{i+7} = u_7 = a \neq b = \tau_{i+14} = w_{i+14} & \quad (\tau_{i+10}\cdots\tau_{i+17} = abbabaab) \\
w_{i+4} = u_4 = b \neq a = \tau_{i+11} = w_{i+11} & \quad (\tau_{i+10}\cdots\tau_{i+17} = baababba) \\
w_{i+9} = \tau_{i+9} = a \neq b = \tau_{i+16} = w_{i+16} & \quad (\tau_{i+10}\cdots\tau_{i+17} = baababba)
\end{aligned}
$$

And for the extended (c2)-wave, too:

$$
\begin{aligned}
w_{i+7} = \tau_{i+7} = a \neq b = \tau_{i+14} = w_{i+14} & \quad (\tau_{i+12}\cdots\tau_{i+19} = abbabaab) \\
w_{i+6} = \tau_{i+6} = b \neq a = \tau_{i+13} = w_{i+13} & \quad (\tau_{i+12}\cdots\tau_{i+19} = baababba)
\end{aligned}
$$

As a result, C2 is false. ■

Lemma 5.11. *Algorithm* ***R5A2*** *won't let Ann lose on a* 5*th power of length* 9.

Proof. Assume we have the word s of length 9 from (5.4) or (5.5). In C1 case we use (5.9) for $k = 4$, so in the same way as in Lemma 5.10 we are able to check that $\xi_2\xi_3 = aa$. Like in Lemma 5.9, $w_{2\hat{i}}w_{2\hat{i}+1} = aa$ implies $\beta_{2\hat{i}-2}w_{2\hat{i}-1}\beta_{2\hat{i}-1} = bbb$ and a contradiction due to $\beta_{2\hat{i}-2} = a$.

For C2 we make the steps as in Lemma 5.9. We apply Proposition 1.9 (squares of length 9) to eliminate 1-letter-wave subcases. Let's fix $w_i = u_0$, where $u = ababbababa$ is an extended (b2)-wave or an extended (c2)-wave, and look for a 9-letter square. The search is unsuccessful for the extended (b2)-wave:

$$\begin{aligned} w_{i+6} = u_6 = b \neq a = \tau_{i+15} = w_{i+15} &\quad (\tau_{i+10}\cdots\tau_{i+17} = abbabaab) \\ w_{i+10} = \tau_{i+10} = a \neq b = \tau_{i+19} = w_{i+19} &\quad (\tau_{i+10}\cdots\tau_{i+21} = abbabaababba) \\ w_{i+9} = \tau_{i+9} = a \neq b = \tau_{i+18} = w_{i+18} &\quad (\tau_{i+10}\cdots\tau_{i+21} = abbabaabbaab) \\ w_{i+7} = u_7 = a \neq b = \tau_{i+16} = w_{i+16} &\quad (\tau_{i+10}\cdots\tau_{i+17} = baababba) \end{aligned}$$

And also for the extended (c2)-wave:

$$\begin{aligned} w_{i+8} = \tau_{i+8} = b \neq a = \tau_{i+17} = w_{i+17} &\quad (\tau_{i+12}\cdots\tau_{i+19} = abbabaab) \\ w_{i+9} = \tau_{i+9} = a \neq b = \tau_{i+18} = w_{i+18} &\quad (\tau_{i+12}\cdots\tau_{i+19} = baababba) \end{aligned}$$

Therefore C2 is not true. ∎

Lemma 5.12. *Algorithm* ***R5A2*** *won't let Ann lose on a* 5*th power of length* 11.

Proof. Assume we have the word s of length 11 from (5.4) or (5.5). In C1 case we use (5.9) for $k = 5$, hence in the same way as in the previous lemmas we are able to find the elements of ξ:

$$\begin{aligned} w_{2i}w_{2i+1} = bb &\quad\Rightarrow\quad \beta_{2i-2}w_{2i-1}\beta_{2i-1} = \xi_4\xi_5 a = aaa \\ w_{2\hat{i}}w_{2\hat{i}+1} = \xi_4 a = aa &\quad\Rightarrow\quad \beta_{2\hat{i}-2}w_{2\hat{i}-1}\beta_{2\hat{i}-1} = \xi_1\xi_2\xi_3 = bbb \\ w_{2\tilde{i}-2}w_{2\tilde{i}-1} = \xi_1\xi_3 = bb &\quad\Rightarrow\quad \beta_{2\tilde{i}-3} = \xi_0 = a \end{aligned}$$

It leads to a contradiction, because $\beta_{2\tilde{i}-4}w_{2\tilde{i}-3}\beta_{2\tilde{i}-3} \neq aaa$.

For C2 we repeat the reasoning from Lemma 5.9. We use Proposition 1.9 (squares of length 11) to eliminate 1-letter-wave subcases. Moreover, the biggest index pointing to the last letter of $w_j w_{j+1}\cdots w_{j+q'}$ modified by an extended (c2)-wave is $j+4$, while $w_{j+5}w_{j+6}\cdots w_{j+26}$ is a square, so the proposition excludes also 4-letter waves.

Let's fix $w_i = u_0$, where $u = ababbababa$ is an extended (b2)-wave and make sure that the wave doesn't produce an 11-letter square:

$$\begin{aligned} w_{i+9} = \tau_{i+9} = a \neq b = \tau_{i+20} = w_{i+20} &\quad (\tau_{i+18}\cdots\tau_{i+21} = abba) \\ w_{i+8} = \tau_{i+8} = b \neq a = \tau_{i+19} = w_{i+19} &\quad (\tau_{i+18}\cdots\tau_{i+21} = baab) \end{aligned}$$

Eventually, C2 does not hold. ∎

Lemma 5.13. *Algorithm **R5A2** won't let Ann lose on a* 5*th power of length* 13*.*

Proof. Assume we have the word s of length 13 from (5.4) or (5.5). In C1 case we use (5.9) for $k = 6$ as in the previous lemmas in order to determine $\xi = aaabbbaa$. It means $w_{2\tilde{i}-4}w_{2\tilde{i}-3}w_{2\tilde{i}-2} = aab$ is a prefix of some extended wave – a contradiction.

Case C2 is similar to the one from Lemma 5.12, so we just need to check whether an extended (b2)-wave introduce a 13-letter square. Let's fix $w_i = u_0$, where $u = ababbababa$ is the extended (b2)-wave.

$$\begin{aligned} w_{i+8} = \tau_{i+8} = b \neq a = \tau_{i+21} = w_{i+21} \quad & (\tau_{i+18} \cdots \tau_{i+21} = abba) \\ w_{i+10} = \tau_{i+10} = a \neq b = \tau_{i+23} = w_{i+23} \quad & (\tau_{i+10} \cdots \tau_{i+25} = abbabaabbaababba) \end{aligned}$$

Note that $\tau_{i+10} \cdots \tau_{i+25}$ could not be $baababbabaababba$, since $\tau_{i+9}\tau_{i+10} \cdots \tau_{i+25}$ would be an overlap. As a result, we contradict C2. ■

Lemma 5.14. *Algorithm **R5A2** won't let Ann lose on a* 5*th power of length* 15*.*

Proof. Assume we have the word s of length 15 from (5.4) or (5.5). In C1 case we use (5.9) for $k = 7$ as in the previous lemmas, which determines $\xi = bbaaabbbaa$ with a contradiction due to $\beta_{2\check{i}-6}w_{2\check{i}-5}\beta_{2\check{i}-4} \neq bbb$.

For C2 we exclude 1-letter and 4-letter waves like in Lemma 5.12. Additionally, the biggest index pointing to the last letter of $w_j w_{j+1} \cdots w_{j+q'}$ modified by an extended (b2)-wave is $j + 6$, while $w_{j+7}w_{j+8} \cdots w_{j+36}$ is a square, hence Proposition 1.9 excludes all possible waves. ■

Lemma 5.15. *Algorithm **R5A2** won't let Ann lose on a* 5*th power of length* 17*.*

Proof. Assume we have the word s of length 17 from (5.4) or (5.5). We know that C2 is false by the same reasoning as in Lemma 5.14. In C1 case we use (5.9) for $k = 8$ as in the previous lemmas and determine $\xi = abbbaaabbbaa$ with a contradiction: $\beta_{2i-8}w_{2i-7}\beta_{2i-7} \neq aaa$. ■

Lemma 5.16. *Algorithm **R5A2** won't let Ann lose on a* 5*th power of length* $2k + 1$ *for* $k \geq 9$*.*

Proof. Assume we have the word s of length $2k + 1$ for $k \geq 9$ from (5.4) or (5.5). We know that C2 is false by the same reasoning as in Lemma 5.14. If C1 holds, then for an arbitrary j there are integers $i, \hat{i}, \tilde{i}, \check{i}$ that the words below are subwords of s^5:

$$\begin{aligned} \beta_{2i-8}w_{2i-7} \cdots \beta_{2i-1}w_{2i}\beta_{2i}w_{2i+1}\beta_{2i+1} &= \xi abbba \\ w_{2\hat{i}-6}\beta_{2\hat{i}-6} \cdots w_{2\hat{i}+1}\beta_{2\hat{i}+1}w_{2\hat{i}+2}\beta_{2\hat{i}+2}w_{2\hat{i}+3} &= \xi abbba \\ \beta_{2\tilde{i}-7}w_{2\tilde{i}-6} \cdots \beta_{2\tilde{i}}w_{2\tilde{i}+1}\beta_{2\tilde{i}+1}w_{2\tilde{i}+2}\beta_{2\tilde{i}+2} &= \xi abbba \\ w_{2\check{i}-7}\beta_{2\check{i}-7} \cdots w_{2\check{i}}\beta_{2\check{i}}w_{2\check{i}+1}\beta_{2\check{i}+1}w_{2\check{i}+2} &= \xi abbba \end{aligned}$$

where ξ is some 14-letter word and

$$\begin{array}{rclrcl} j+q'-\lceil q\rceil+8\leq & 2i & \leq j+\lfloor q\rfloor-1, & j+6\leq & 2\hat{i} & \leq j+q'-3,\\ j+q'-\lceil q\rceil+7\leq & 2\tilde{i} & \leq j+\lfloor q\rfloor-2, & j+7\leq & 2\check{i} & \leq j+q'-2 \end{array}$$

The values of $i,\hat{i},\tilde{i},\check{i}$ are the same as in (5.10) for odd k and as in (5.11) for even k. By applying the same technique as in the preceding lemmas: $\xi = aaabbbaaabbbaa$.

We have $w_{2\hat{i}-6}w_{2\hat{i}-5}\cdots w_{2\hat{i}+3} = aabaabaaba$. Observe subwords aab at positions $2\hat{i}-6$ and $2\hat{i}$. Both positions are even, but the distance between the subwords is not dvisible by 4 – it contradicts Remark 5.5 and ends the proof. ■

Finally, we completed the fragment of the section devoted to odd 5th powers. The part related to even 5th powers consists of three main cases, only three special cases and an auxilliary algorithm for turning output words into τ.

Lemma 5.17. *Algorithm **R5A2** won't let Ann lose on a* 5*th power of length* 2*.*

Proof. We only need to justify no trivial 5th powers in any output word. Indeed, τ is overlap-free with at most trivial squares, and extended waves introduce at most trivial cubes. ■

Lemma 5.18. *Algorithm **R5A2** won't let Ann lose on a* 5*th power of length* $4k+2$ *for* $k \geq 1$*.*

Proof. Assume we have the word s of length $4k+2$ for $k \geq 1$ from (5.6) or (5.7). It means that $x = w_j w_{j+1}\cdots w_{j+10k+4}$ is a 5th power of length $2k+1$. At first, notice that at least one of these words has to be entirely inside x:
(1) (a)-wave with first two letters of (a)-wave gap: aab;
(2) (b1)-wave with first two letters of (b1)-wave gap: aaa;
(3) (c1)-wave with first two letters of (c1)-wave gap: aaa;
(4) four first letters of (b2)-wave: $baab$;
(5) five last letters of (b2)-wave: $aabab$;
(6) three last letters of (c2)-wave: aab.

Otherwise, we obtain $x_4x_5\cdots x_{10k+1} = \tau_{j+4}\tau_{j+5}\cdots\tau_{j+10k+1}$, because at most first four or at most last three elements of x could still be parts of waves without violating the negation of the previous statement. Hence we receive an overlap $x_4x_5\cdots x_{4k+6}$ for $k \geq 1$, which contradicts Theorem 1.6.

Assume word (1) is located at some position i in x and simultaneously at even position in w (first a is the wave). Then either second aab is located at position $i-4k-2$ if $i \geq 4k+2$, or at position $i+4k+2$ if $i < 4k+2$. However, the distance between the first and the second aab is not divisible by 4, contradicting Remark 5.5.

For words (2) and (3) if we know that one of them is located at i in x, then the second aaa is at $i-2k-1$ for $i \geq 2k+1$ or at $i+2k+1$ for $i < 2k+1$. Since aaa can be located only at even positions in w, it contradicts Remark 5.5 too.

When words (4), (5) or (6) are subwords of x, then x has a subword aab located at some index i. Consequently, the second aab is at $i-2k-1$ for $i \geq 2k+1$ or at $i+6k+3$ for $i < 2k+1$, and the third aab is at $i+2k+1$ for $i \leq 8k+1$ or at $i-6k-3$ for $i > 8k+1$. The first aab is located at odd position in w, whereas the second and third one – at even positions. The distance between the second and the third aab is not divisible by 4, which contradicts Remark 5.5 one more time.

At the end, it is straightforward to check that all considered words are entirely inside x for $k \geq 1$. ■

Lemma 5.19. *Algorithm **R5A2** won't let Ann lose on a* 5*th power of length* 4.

Proof. Assume we have the word s of length 4 from (5.6) or (5.7). It means that $w_j w_{j+1} \cdots w_{j+9}$ is a 5th power of length 2. For C1 we take r from (5.8). If $r < 7$ then:

$$w_{j+r+2}\beta_{j+r+2}w_{j+r+3} = w_{j+r}\beta_{j+r}w_{j+r+1} = abb$$

$\tau_{j+r+2} = b$ means that $\tau_{j+r+3} = a$, because there is no 2-letter segment bb in τ. Although $w_{j+r+2}w_{j+r+3}$ seems to be the beginning of (b2)-wave or (c2)-wave, $\beta_{j+r+2} \neq a$ contradicts this statement.

Thus $r = 7$ and consequently $w_j w_{j+1} \cdots w_{j+9} = bababababa$ with odd j. Since $\tau_{j+4}\tau_{j+5} \cdots \tau_{j+8}$ is not an overlap, at least one letter from $w_{j+4}, w_{j+5}, w_{j+6}$ is modified by some wave. It's not a 1-letter wave, since such wave produces 2-letter segments aa and bb. Consider other subcases to obtain a contradiction:
(c2)-wave $w_{j+3}w_{j+4}w_{j+5}w_{j+6}$ – impossible: $w_{j+3}w_{j+4} \neq ba$;
(b2)-wave $w_{j+1}w_{j+2}w_{j+3}w_{j+4}w_{j+5}w_{j+6}$ – impossible: $w_{j+1}w_{j+2} \neq ba$;
(c2)-wave $w_{j+1}w_{j+2}w_{j+3}w_{j+4}$ – impossible: $w_{j+1}w_{j+2} \neq ba$;
(b2)-wave $w_{j-1}w_j w_{j+1}w_{j+2}w_{j+3}w_{j+4}$ – impossible: $w_j \neq a$.

For C2 we observe the biggest index pointing to the last letter of $w_j w_{j+1} \cdots w_{j+9}$ modified by an extended wave except for (b2)-wave is $j+4$, while $w_{j+5}w_{j+6} \cdots w_{j+9}$, a subword of τ, is an overlap. It contradicts overlap-freeness of τ.

In the remaining subcase we should check if an extended (b2)-wave $u = ababbababa$ introduces a 5th power $w_j w_{j+1} \cdots w_{j+9}$ that $i+1 \leq j \leq i+7$, where $w_i = u_0$.

$$\begin{aligned}
w_{i+3} &= u_3 = b \neq a = u_5 = w_{i+5} \\
w_{i+10} &= \tau_{i+10} = a \neq b = \tau_{i+12} = w_{i+12} \quad (\tau_{i+10} \cdots \tau_{i+13} = abba) \\
w_{i+10} &= \tau_{i+10} = b \neq a = \tau_{i+12} = w_{i+12} \quad (\tau_{i+10} \cdots \tau_{i+13} = baab)
\end{aligned}$$

The inequalities above are sufficient to negate C2 also for extended (b2)-waves. ■

Lemma 5.20. *Algorithm **R5A2** won't let Ann lose on a* 5*th power of length* $8k+4$ *for* $k \geq 1$.

Proof. Assume we have the word s of length $8k+4$ for $k \geq 1$ from (5.6) or (5.7). It means that $x = w_j w_{j+1} \cdots w_{j+20k+9}$ is a 5th power of length $4k+2$. At first, notice

that at least one of these words has to be entirely inside x:
(1) (a)-wave with first two letters of (a)-wave gap: *aab*;
(2) (b1)-wave with first two letters of (b1)-wave gap: *aaa*;
(3) (c1)-wave with first two letters of (c1)-wave gap: *aaa*;
(4) (b2)-wave with the minimal (b2)-wave gap: *abbababa*;
(5) (c2)-wave with the minimal (c2)-wave gap: *abbababa*.

Otherwise, we are sure that $x_5x_6\cdots x_{20k+2} = \tau_{j+5}\tau_{j+6}\cdots\tau_{j+20k+2}$, because at most first five or some of at most last seven elements of x could be signs from waves without violating the negation of the previous statement. Hence we receive an overlap $x_5x_6\cdots x_{8k+9}$ for $k \geq 1$, which contradicts Theorem 1.6.

We lead the same reasoning as in Lemma 5.18 to show that word (1) is not a part of x. If word (2) or (3) is a subword of x located at position i, then the second *aaa* appears either at $i-8k-4$ for $i \geq 8k+4$, or at $i+8k+4$ for $i < 8k+4$. Let's denote the distance between the first and the second *aaa* by Δ. Obviously, we have $\Delta \equiv 4 \pmod 8$. Remark 5.5 implies $\Delta \equiv d \pmod 8$ where $d \in \{0,2,6\}$, and we receive a contradiction.

Assume we found word (4) at position i in x. It means that $j+i \equiv 0 \pmod 8$. Consequently, there is also the second word *abbababa* in x at position $\hat{\imath}$ satisfying:

$$\hat{\imath} = \begin{cases} i-8k-4 & \textit{if} \quad k \equiv 0 \pmod 2 \quad \textit{and} \quad i \geq 8k+4 \\ i+4k+2 & \textit{if} \quad k \equiv 0 \pmod 2 \quad \textit{and} \quad i < 8k+4 \\ i-4k-2 & \textit{if} \quad k \equiv 1 \pmod 2 \quad \textit{and} \quad i \geq 4k+2 \\ i+8k+4 & \textit{if} \quad k \equiv 1 \pmod 2 \quad \textit{and} \quad i < 4k+2 \end{cases}$$

Similarly, when word (5) appears at position i in x, we obtain $j+i \equiv 6 \pmod 8$ and we consider the second *abbababa* located at position $\hat{\imath}$ that:

$$\hat{\imath} = \begin{cases} i-4k-2 & \textit{if} \quad k \equiv 0 \pmod 2 \quad \textit{and} \quad i \geq 4k+2 \\ i+8k+4 & \textit{if} \quad k \equiv 0 \pmod 2 \quad \textit{and} \quad i < 4k+2 \\ i-8k-4 & \textit{if} \quad k \equiv 1 \pmod 2 \quad \textit{and} \quad i \geq 8k+4 \\ i+4k+2 & \textit{if} \quad k \equiv 1 \pmod 2 \quad \textit{and} \quad i < 8k+4 \end{cases}$$

After small calculations we receive in each subcase either $j+\hat{\imath} \equiv 2 \pmod 8$, or $j+\hat{\imath} \equiv 4 \pmod 8$. Both equivalences lead to contradictions:

$j+\hat{\imath} \equiv 2 \pmod 8$: there is a 4-letter segment *baba* at $j+\hat{\imath}+4$; according to Remark 5.6, it must be preceded by *abba* in case of (b2)-wave or followed by *abab* in case of (c2)-wave.

$j+\hat{\imath} \equiv 4 \pmod 8$: there is a 4-letter segment *baba* at $j+\hat{\imath}+8$; according to Remark 5.6, it can be only preceded by *abab* (in case of (c2)-wave).

It is easy to check that all the second words are entirely inside x for $k \geq 1$. ■

```
Require: word is at position p in an output word, where p ≡ pos (mod 8)
function InterpretAsTau(word, pos)
    c ← 8 − pos (mod 4)                  ▷ position of the first 4-letter segment in word
    len ← length(word)
    result ← word
    while c + 3 < len do
        if word[c...c+3] ∈ {0100, 1011} then               ▷ (a)-wave or (c1)-wave
            result[c+2] ← 1 − word[c+2]                     ▷ abaa ⇒ abba
        else if word[c...c+3] ∈ {0001, 1110} then           ▷ (b1)-wave
            result[c]     ← 1 − word[c]                     ▷ aaab ⇒ baab
        else if word[c...c+3] ∈ {0101, 1010} then
            if c + 7 < len and c + pos ≡ 4 (mod 8) then
                if word[c...c+7] ∈ {01011010, 10100101} then    ▷ (c2)-wave
                    result[c+2] ← 1 − word[c+2]             ▷ abab ⇒ abba
                    result[c+3] ← 1 − word[c+3]
                    result[c+4] ← 1 − word[c+4]             ▷ baba ⇒ abba
                    result[c+5] ← 1 − word[c+5]
                else                                        ▷ (b2)-wave
                    if c ≥ 4 then                           ▷ the previous segment
                        result[c−4] ← 1 − word[c−4]         ▷ baab ⇒ abba
                        result[c−3] ← 1 − word[c−3]
                        result[c−2] ← 1 − word[c−2]
                        result[c−1] ← 1 − word[c−1]
                    end if
                    result[c]     ← 1 − word[c]             ▷ abab ⇒ baab
                    result[c+1] ← 1 − word[c+1]
                end if
            end if        ▷ c + pos ≡ 0 (mod 8) ⇒ (c2)-wave one iteration before
        end if
        c ← c + 4                             ▷ moving to the next 4-letter segment
    end while
    return result    ▷ note: some first and last letters might be uninterpreted
end function
```

We start the part of the section devoted to the proofs for 5th powers of length divisible by 8 by revealing a function InterpretAsTau: $\{0,1\}^* \times \mathbb{Z}_8 \to \{0,1\}^*$ as an algorithm based on Remark 5.6 for changing subwords of output words (first parameter), with an auxilliary information about the position of the first 8-letter segment (second parameter), back into subwords of τ.

The function does not receive the exact position of subwords inside the output words, so it has to analyse their content to give the right result. InterpretAsTau replaces the full 4-letter segments of output words with 4-letter segments of τ (and, as we will see, not all of them), so only a subword of the first parameter of length divisible by 4 is interpreted – other symbols are left unchanged.

Remark 5.6 shows that Algorithm **R5A2** introduces three new types of 4-letter segments to already known *abba* from τ. InterpretAsTau handles them as follows:

1. *abaa* – always directly interpreted as *abba* in line 7.
2. *aaab* – always directly interpreted as *baab* in line 9.
3. *abab* – either the consequence of (b2)-wave (lines 18–25), or (c2)-wave (lines 13–16). In order to decide we need to examine the succeeding 4-letter segment in line 12. The verification is valid because if *abab* appears at position divisible by 8 in some output word, then (from the remark) it must be the second segment related to (c2)-wave (see lines 11 and 27) – we change it immediately. Interestingly, the case of (b2)-wave is the only case in which the function has to modify the segment before the current one and in which *abba* is changed.

To summarize, InterpretAsTau unambigously translates the segments specific to output words, yet in case of *abab* it needs to preview the next segment. When it's impossible, it lefts one ((c2)-wave) or two ((b2)-wave) last full segments of $word$ unchanged. Moreover, if the first full segment of $word$ is *abab* and $pos \in \{0, 5, 6, 7\}$, then the function skips the segment – there is no iteration before (the comment in line 27).

Lemma 5.21. *Algorithm **R5A2** won't let Ann lose on a* 5*th power of length* 8.

Proof. Assume we have the word s of length 8 from (5.6) or (5.7). It means that $x = w_j w_{j+1} \cdots w_{j+19}$ is a 5th power of length 4. As a consequence, x contains at least four full identical 4-letter segments y. Let's look closely at each possibility of y:

$y = abaa$: InterpretAsTau$(y^4, p) = (abba)^4$, where $p \in \{0, 4\}$. The whole y^4 is interpreted, yet τ has no 4th powers – y^4 cannot be a subword of x.

$y = aaab$: InterpretAsTau$(y^4, p) = (baab)^4$, where $p \in \{0, 4\}$. Hence y^4 is not a subword of x.

$y = abab$: Remark 5.6 indicates that y must be an effect of (b2)-wave or (c2)-wave, and both of them introduce a pair of different 4-letter segments. There is no way to construct y^4 from them; we are able to create at most y^2 by setting together segments from two (c2)-waves: $baba\underline{abababab}baba$.

$y = abba$: InterpretAsTau$(y^4, p) = (abba)^4$, where $p \in \{0, 4\}$. The last 4-letter segment might be a part of (b2)-wave (if $p = 4$) and it could be left uninterpreted due to the lack of the next full segments. However, we are sure that first three segments are interpreted correctly, which means no y^4 in x because of cubefreeness of τ.

We did not find any valid y^4, which contradicts the existence of x. ■

Lemma 5.22. *Algorithm **R5A2** won't let Ann lose on a* 5*th power of length* $8k$ *for* $k \geq 2$.

Proof. Assume we have the word s of length $8k$ from (5.6) or (5.7). It means that $x = w_j w_{j+1} \cdots w_{j+20k-1}$ is a 5th power of length $4k$. Thus x contains at least $5k-1$ full 4-letter segments.

Let's think about the possible effect of applying the function InterpretAsTau to x. Observe that the last two full segments could be uninterpreted in case of (b2)-wave. The similiar situation may happen to the first full segment of x when it is the second segment related to (c2)-wave. Therefore, we obtain $5k-4$ full segments interpreted by the function, which leads to at least $3k$ full segments for $k \geq 2$. Notice the word built on these $3k$ 4-letter segments is a cube of length $4k$, which is translated by InterpretAsTau into another cube. Hence we receive a contradiction, because τ doesn't contain 3rd powers. ■

At last:

Theorem 5.23. *There exists a strategy with finite description for Ann that allows her to win the* 5*th-power-free game of any length on* 2 *letters.*

Proof. We covered all the cases in Lemmas 5.7–5.22 and therefore proved that Algorithm **R5A2** is the right strategy for Ann. ■

Squarefree colourings of line arrangements

The contents of the chapter were originally published in our paper [21]. This part of the thesis is very autonomous with its own vocabulary, theorems and conjectures, yet it is still clearly connected to the topic of nonrepetitive sequences.

Let's consider *squarefree colourings* of graphs in which a colour sequence of every simple path is squarefree. The least number of colours in such colouring of a graph G is denoted by $\pi(G)$ and called the *Thue chromatic number* of G. It was proved that $\pi(G)$ is bounded for graphs of bounded degree [2] and for graphs of bounded treewidth [5, 25], but the following conjecture remains open (see [16]):

Conjecture 6.1. *There is a constant C such that every planar graph G satisfies $\pi(G) \leq C$.*

In this chapter we study a geometric variant of squarefree colourings inspired by this conjecture. Let L be a *line arrangement* consisting of a finite set of lines in the plane. Let $P = P(L)$ denote the set of all intersection points of these lines. A *squarefree colouring* of L is a colouring of the set P such that a sequence of colours determined by consecutive points on every line in L is squarefree.

6.1 The main result

We start with recalling a definition of homomorphism of edge coloured graphs. We say that an edge coloured graph G has a *homomorphic embedding* into another edge coloured graph H if there is a function $h : V(G) \to V(H)$ such that for every pair of adjacent vertices $u, v \in V(G)$, their images $h(u)$ and $h(v)$ are adjacent in H, and colour of the edge $h(u)h(v)$ is the same as colour of the edge uv. The following lemma comes from [3].

Lemma 6.2 (Alon, Marshall 1998)**.** *Let k be a positive integer. There exists a graph H_k on at most $5k^4$ vertices with k-coloured edges such that every planar graph G whose edges are coloured arbitrarily with k colours embeds homomorphically into H_k.*

The proof of this result is based on the fact that planar graphs have bounded *acyclic chromatic number* $\chi_a(G)$, defined as the least number of colours in a proper vertex colouring of G with no 2-coloured cycles. By a famous theorem of Borodin [11] every planar graph G satisfies $\chi_a(G) \leq 5$, which is best possible.

Theorem 6.3 (Grytczuk, Kosiński, Zmarz 2015). *Every line arrangement has a squarefree colouring using at most* 405 *colours.*

Proof. Let L be a finite set of lines in the plane and let P be the set of all intersection points determined by L. Consider a graph $G = G(L)$ on the vertex set P with two points p and q adjacent if and only if there is a line in L containing both of them, and there is no other point of P placed between p and q on this line. Clearly G is a planar graph.

A path in G whose vertices are colinear will be called a *straight path.* Using Proposition 1.16 we may colour the edges of G by three colours such that no straight path contains a pattern of the form bxb: just take sufficiently long sequence s with this property and colour the edges of each straight path consecutively, accordingly to s. No conflicts between paths may arise as they are edge disjoint. Let's denote by $c : E(G) \to \{1, 2, 3\}$ any colouring satisfying this property.

Now we apply Lemma 6.2 to G with edge colouring c. Let H_3 and $h : P \to V(H_3)$ be a graph, and a homomorphism satisfying assertion of the lemma, respectively. We claim that h is a squarefree colouring of L. Suppose that this is not the case and let q be a squarely coloured straight path in G of length $2k$ with edges denoted by $e_i = q_i q_{i+1}$, $0 \leq i \leq 2k-2$. This means that $h(q_i) = h(q_{i+k})$ for $0 \leq i < k$. Thus, by the edge colour preservation property of h, we get: $c(e_i) = c(e_{i+k})$ for $0 \leq i \leq k-2$. However, it implies that the colour pattern of q in colouring c has the form bxb, with $b = c(e_0)c(e_1) \cdots c(e_{k-2})$ and $x = c(e_{k-1})$. It contradicts the property of colouring c and finishes the proof. ∎

Moreover, Lemma 6.2 is a special case of the more general result from [3]:

Theorem 6.4 (Alon, Marshall 1998). *For every pair of integers k and r there exists a graph $H_{k,r}$ with edges coloured by k colours such that every k-edge coloured graph G with $\chi_a(G) \leq r$ embeds homomorphically into $H_{k,r}$. Moreover, the least number of vertices in such graph $H_{k,r}$ is bounded by rk^{r-1}.*

By this result and Proposition 1.16 the proof of Theorem 6.3 generalizes easily, giving the statement below.

Theorem 6.5. *Let G be a graph with $\chi_a(G) \leq r$ whose edge set is decomposed arbitrarily into simple paths $Q_0, Q_1, \ldots, Q_{m-1}$. Then there exist a vertex colouring of G using at most $r3^{r-1}$ colours such that the sequence of colours on every path Q_i is squarefree.*

6.2 Squarefree colouring of the plane

The famous Hadwiger-Nelson problem asks for the *chromatic number of the plane* $\chi(\mathbb{R}^2)$, defined as the least number of colours needed to colour the plane such that every pair of points at distance one apart is coloured differently (see [32]). We formulate a natural analogous question in the spirit of squarefree colourings of line arrangements.

A finite sequence of points $P_0, P_1, \cdots, P_{m-1}$ in the plane is called *nice* if the points are colinear and the distance between each pair P_i and P_{i+1} is one. A colouring of the plane is *squarefree* if a sequence of colours determined by every nice sequence of points is squarefree. Let $\pi(\mathbb{R}^2)$ denote the least number of colours needed for a squarefree colouring of the plane. We prove below that this number is finite. As before we will need the following simple consequence of the Thue result, related to two-sided infinite words [34]. Interestingly, the bound of 6 is optimal [15].

Proposition 6.6. *There exists a* 6*-colouring of the integers such that every finite sequence of integers whose consecutive terms differ by at most* 2 *is coloured squarefreely.*

Proof. Let $A = \{a, b, c\}$ and $A' = \{a', b', c'\}$ be two disjoint sets of colours. By the result of Thue there is a colouring $f : \mathbb{Z} \to \{a, b, c\}$ such that every subword of integers is squarefree. Let $f' : \mathbb{Z} \to \{a', b', c'\}$ be a true copy of colouring f. Define a new colouring $g : \mathbb{Z} \to \{a, b, c, a', b', c'\}$ by shuffling f and f', that is, by $g(2n) = f(n)$ and $g(2n + 1) = f'(n)$ for every $n \in \mathbb{Z}$. It is not hard to see that g satisfies the assertion of the lemma. Indeed, let $m = m_0 m_1 \cdots m_{2k-1}$ be any sequence of integers satisfying $m_{i+1} - m_i \in \{1, 2\}$. Let $m^{(e)}$ and $m^{(o)}$ be subsequences of m consisting of even and odd terms, respectively. It is clear that both are arithmetic progressions of difference 2, and at least one of them is nonempty, say $m^{(e)}$. Suppose now that the sequence of colours on m is a square. It follows that restriction of g to $m^{(e)}$ must also form a square. But this contradicts squarefreeeness of f and completes the proof. ∎

With this lemma at hand we may prove the following result.

Theorem 6.7 (Grytczuk, Kosiński, Zmarz 2015)**.** $\pi(\mathbb{R}^2) \le 36$.

Proof. Let g be a colouring of the integers satisfying the assertion of Proposition 6.6. We extend g to a colouring of the real line as follows. First we split $\mathbb{R}$ into half open intervals $I_n = [a_n, b_n)$, ($n \in \mathbb{Z}$), each of length $1/\sqrt{2}$. Then we colour each point of I_n with $g(n)$. Next we define the product colouring h of the plane by $h(x, y) = (g(x), g(y))$. We claim that h is a squarefree colouring of $\mathbb{R}^2$. To see this let $S = P_1 P_2 \dots P_n$ be a nice sequence of points, with $P_i = (x_i, y_i)$. Define a function $p : \mathbb{R} \to \mathbb{Z}$ by $p(x) = n$ if $x \in I_n$. Next consider two sequences $S_x = p(x_1)p(x_2) \cdots p(x_n)$ and $S_y = p(y_1)p(y_2) \cdots p(y_n)$. It is not hard to see that at least one of the sequences S_x or S_y has gaps precisely in the set $\{1, 2\}$ (or in $\{-1, -2\}$). This is because colouring h determines a tiling of $\mathbb{R}^2$ into squares of diameter slightly less than one and therefore any line cannot "jump" more than two squares. It follows by the property of g that a colour sequence of S_x or S_y is squarefree, which implies the same for the colour sequence of S. The proof is complete. ■

6.3 Problems and remarks

We naturally expect that the bound of 405 from Theorem 6.3 is not optimal. One possible way of lowering it is to improve Borodin's theorem for planar graphs arising from line arrangements. Indeed, we could not find any such graph with $\chi_a(G) = 5$. On the other hand, the arrangement of six lines determined by four sides and two diagonals of the square gives a graph G with $\chi_a(G) = 4$.

Conjecture 6.8. *Every planar graph G arising from line arrangement satisfies $\chi_a(G) \leq 4$.*

If true, the conjecture would imply that every line arrangement has a squarefree colouring using at most 108 colours. This still does not sound like the best possible bound. We propose the following (risky) conjecture.

Conjecture 6.9. *Every line arrangement has a squarefree colouring with at most 4 colours.*

The conjecture is optimal as is seen in arrangement of four lines intersecting at one point and the fifth line intersecting other ones at four different points. For the problem of colouring the plane, the upper bound of 36 given in Theorem 6.7 is improved to 18 in [35]. However, this time guessing that $\pi(\mathbb{R}^2) = 4$ seems definitely too risky (though it is not completely ruled out). More results regarding nonrepetitive colourings of line arrangements are presented in [14] and [35].

Let us conclude with a conjecture generalizing our results. Recall that a *geometric graph* is a graph drawn on the plane such that each vertex corresponds to a point and every edge is a closed line segment connecting two vertices but not passing through a third. A *straight path* in a geometric graph G is a path whose vertices are colinear. A *straight squarefree colouring* of a geometric graph is a colouring of its vertices such that the sequence of colours on any straight path is squarefree. Let $\overline{\pi}(G)$ denote the least number of colours needed in such colouring of G.

Conjecture 6.10. *There is a function $f : \mathbb{N} \to \mathbb{N}$ such that every geometric graph G satisfy $\overline{\pi}(G) \leq f(\chi(G))$.*

Clearly the chromatic number $\chi(G)$ is a lower bound for $\overline{\pi}(G)$. For graphs arising from line arrangements $\chi(G) \leq 4$ by the Four Colour Theorem. For unit distance graphs we have $\chi(G) \leq 7$. Another example in favour of our conjecture is given by the geometric (visibility) graph $\mathcal{V}(\mathbb{Z}^2)$ generated by integer lattice points in the plane (two lattice points are connected if there is no other lattice point on a line segment between them). It is not hard to see that that $\chi(\mathcal{V}(\mathbb{Z}^2)) = 4$, and it was proved by Carpi [12] that $\overline{\pi}(\mathcal{V}(\mathbb{Z}^2)) \leq 16$.

Conclusion & open problems

In order to summarize our results for "standard" nonrepetitive games let's look at Table 7.1. The rows in which upper and lower bounds are different naturally trigger open problems. In case of the squarefree game we can ask even two questions:

Problem 7.1. *For the squarefree game, what is the minimal cardinality of an alphabet that guarantees a winning strategy for Ann?*

From Theorems 1.3 and 1.5 we know that the right number might be 4, 5, or 6. Finding an algorithmic strategy that works for less than 8 symbols would be interesting as well:

Problem 7.2. *For the squarefree game, what is the minimal cardinality of an alphabet that guarantees a winning strategy of finite description for Ann?*

It would be also nice to know whether the answer in these two variants above is the same or not.

Game type	**Alphabet size for an algorithmic strategy**			
	Lower bound		Upper bound	
	Value	Reference	Value	Reference
(non-trivial-)squarefree	4	Theorem 1.5	8	Theorem 3.6
overlap-free	4	Corollary 4.5	4	Corollary 4.4
cubefree	3	Theorem 5.2	4	Corollary 4.4
non-trivial-cubefree	2	—	3	Corollary 2.11
4th-power-free	3	Theorem 5.2	3	Corollary 2.12
non-trivial-4th-power-free	2	—	2	Corollary 2.18
5th-power-free	2	—	2	Theorem 5.23

Table 7.1: Currently known minimal sizes of an alphabet which allow Ann to win non-sparse nonrepetitive games. Note that a unary alphabet always results in Ann's loss.

Perhaps it is possible to use techniques from Theorems 3.7 or 3.12, and somehow eliminate the small squares. Inspired by Entringer, Jackson and Schatz (see Theorems 1.17 and 1.18), we may also wonder about an analogous game over a 2-letter alphabet.

Problem 7.3. *Consider a squarefree game over a binary alphabet in which Ann must avoid squares bigger than k only. Is there a k that allows her to win in such game independently of the size of the board?*

The remaining gaps in the bounds from Table 7.1 lead to formulate the following problems:

Problem 7.4. *Does Ann have a winning strategy in the cubefree game over a ternary alphabet?*

Problem 7.5. *Does Ann have a winning strategy in the non-trivial-cubefree game over a binary alphabet?*

It's time to change the topic to sparse nonrepetitive games. We know only a small number of facts related to them, and these facts entirely refer to overlap-free and squarefree cases.

Problem 7.6. *For the sparse mth-power-free ($m \geq 3$) game and its non-trivial variant, does Ann have a winning strategy for the same sizes of an alphabet as in the "standard" games of the same type?*

Besides, Theorems 4.2 and 4.3 don't exhaust the subject of the overlap-free case:

Problem 7.7. *For the sparse overlap-free game of even length one can consider a variant in which Ann makes the first move. Does Ann have a winning strategy in this variant or not?*

Let us conclude the thesis with the most mysterious Thue-type game.

Problem 7.8. *Ann and Ben are filling holes of a partial word with letters (like in the sparse overlap-free game). This time none of them, neither Ann, nor Ben, is allowed to create a square. So, the game stops if either there are no more holes to be filled, or if filling any existing hole would produce a square. Ann wins in the former case, Ben in the latter. Who wins this game over a finite alphabet?*

References

[1] J.-P. Allouche and J. Shallit. *Automatic Sequences. Theory, Applications, Generalizations.* Cambridge University Press, 2003.

[2] N. Alon, J. Grytczuk, M. Hałuszczak, and O. Riordan. Non-repetitive colorings of graphs. *Random Structures Algorithms*, 21:336–346, 2002.

[3] N. Alon and T. H. Marshall. Homomorphisms of edge-colored graphs and Coxeter groups. *Journal of Algebraic Combinatorics*, 8:5–13, 1998.

[4] N. Alon and J. H. Spencer. *The probabilistic method, third edition.* Wiley-Interscience Series in Discrete Mathematics and Optimization. Wiley, Hoboken, NJ, 2008.

[5] J. Barát and P. P. Varjú. On square-free vertex colorings of graphs. *Studia Scientiarum Mathematicarum Hungarica*, 44:411–422, 2007.

[6] J. Berstel. Axel Thue's papers on repetitions in words: a translation. In *Publications du Laboratoire de Combinatoire et d'Informatique Mathématique*, volume 20. Université du Québec à Montréal, 1995.

[7] J. Berstel and D. Perrin. The origins of combinatorics on words. *European Journal of Combinatorics*, 28:996–1022, 2007.

[8] F. Blanchet-Sadri, I. Choi, and R. Mercaş. Avoiding large squares in partial words. *Theoretical Computer Science*, 412(29):3752–3758, 2011.

[9] F. Blanchet-Sadri, R. Mercaş, A. Rashin, and E. Willett. Periodicity algorithms and a conjecture on overlaps in partial words. *Theoretical Computer Science*, 443:35–45, 2012.

[10] F. Blanchet-Sadri, R. Mercaş, and G. Scott. A generalization of Thue freeness for partial words. *Theoretical Computer Science*, 410(8-10):793–800, 2009.

[11] O. V. Borodin. On acyclic colorings of planar graphs. *Discrete Mathematics*, 25:211–236, 1979.

[12] A. Carpi. Multidimensional unrepetitive configurations. *Theoretical Computer Science*, 56:233–241, 1988.

[13] J. Currie. Pattern avoidance: themes and variations. *Theoretical Computer Science*, 339:7–18, 2005.

[14] M. Dębski, J. Grytczuk, U. Pastwa, B. Pilat, J. Sokół, M. Tuczyński, P. Wenus, and K. Węsek. On avoiding r-repetitions in $\mathbb{R}^2$. *Electronic Notes in Discrete Mathematics*, 61:331–337, 2017.

[15] M. Dębski, U. Pastwa, and K. Węsek. Grasshopper avoidance of patterns. *The Electronic Journal of Combinatorics*, 23:P4.17, 2016.

[16] V. Dujmović, F. Frati, G. Joret, and D. R. Wood. Nonrepetitive colourings of planar graphs with $O(\log n)$ colours. *The Electronic Journal of Combinatorics*, 20:P51, 2013.

[17] R. C. Entringer, D. E. Jackson, and J. A. Schatz. On nonrepetitive sequences. *Journal of Combinatorial Theory, Series A*, 16(2):159–164, 1974.

[18] A. S. Fraenkel and J. Simpson. How many squares must a binary sequence contain? *The Electronic Journal of Combinatorics*, 2(#R2), 1995.

[19] J. Grytczuk. Thue type problems for graphs, points, and numbers. *Discrete Mathematics*, 308:4419–4429, 2008.

[20] J. Grytczuk, K. Kosiński, and M. Zmarz. How to play Thue games. *Theoretical Computer Science*, 582:83–88, 2015.

[21] J. Grytczuk, K. Kosiński, and M. Zmarz. Nonrepetitive colorings of line arrangements. *European Journal of Combinatorics*, 51:275–279, 2016.

[22] J. Grytczuk, J. Kozik, and P. Micek. New approach to nonrepetitive sequences. *Random Structures and Algorithms*, 42(2):214–225, 2013.

[23] T. Harju and D. Nowotka. Binary words with few squares. *Bulletin of the EATCS*, 89:164–166, 2006.

[24] K. Kosiński, R. Mercaş, and D. Nowotka. Corrigendum to "A note on Thue games" [Inf. Process. Lett. 118 (2017) 75–77]. *Information Processing Letters*, 130:63–65, 2018.

[25] A. Kündgen and M. J. Pelsmajer. Nonrepetitive colorings of graphs of bounded treewidth. *Discrete Mathematics*, 308:4473–4478, 2008.

[26] M. Lothaire. *Combinatorics on Words*. Addison-Wesley, Reading, MA, 1983.

[27] R. Mercaş and D. Nowotka. A note on Thue games. *Information Processing Letters*, 118:75–77, 2017.

[28] R. A. Moser and G. Tardos. A constructive proof of the general Lovász Local Lemma. *Journal of the ACM*, 57:11:1–11:15, 2010.

[29] J. Nešetřil and P. O. de Mendez. *Sparsity*, volume 28 of *Algorithms and Combinatorics*. Springer, 2012.

[30] W. Pegden. Highly nonrepetitive sequences: Winning strategies from the Local Lemma. *Random Structures and Algorithms*, 38(1-2):140–161, 2011.

[31] N. Rampersad, J. Shallit, and M. Wang. Avoiding large squares in infinite binary words. *Theoretical Computer Science*, 339(1):19–34, 2005.

[32] A. Soifer. *The mathematical coloring book*. Springer, 2009.

[33] A. Thue. Über unendliche Zeichenreihen. *Norske Vid. Selsk. Skr. I, Mat. Nat. Kl. Christiania*, 7:1–22, 1906.

[34] A. Thue. Über die gegenseitige Lage gleicher Teile gewisser Zeichenreihen. *Norske Vid. Selsk. Skr. I, Mat. Nat. Kl. Christiania*, 1:1–67, 1912.

[35] P. Wenus and K. Węsek. Nonrepetitive and pattern-free colorings of the plane. *European Journal of Combinatorics*, 54:21–34, 2016.